International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.4103

(02/2022)

SERIES Q: SWITCHING AND SIGNALLING, AND ASSOCIATED MEASUREMENTS AND TESTS

Protocols and signalling for peer-to-peer communications

Hybrid peer-to-peer communications: Overlay management protocol

Recommendation ITU-T Q.4103

ITU-T Q-SERIES RECOMMENDATIONS

SWITCHING AND SIGNALLING, AND ASSOCIATED MEASUREMENTS AND TESTS

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4, 5, 6, R1 AND R2	Q.120–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.799
Q3 INTERFACE	Q.800–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
SPECIFICATIONS OF SIGNALLING RELATED TO BEARER INDEPENDENT CALL CONTROL (BICC)	Q.1900–Q.1999
BROADBAND ISDN	Q.2000–Q.2999
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR THE NGN	Q.3000–Q.3709
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR SDN	Q.3710–Q.3899
TESTING SPECIFICATIONS	Q.3900–Q.4099
PROTOCOLS AND SIGNALLING FOR PEER-TO-PEER COMMUNICATIONS	**Q.4100–Q.4139**
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2020	Q.5000–Q.5049
COMBATING COUNTERFEITING AND STOLEN ICT DEVICES	Q.5050–Q.5069

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

Recommendation ITU-T Q.4103

Hybrid peer-to-peer communications: Overlay management protocol

Summary

Recommendation ITU-T Q.4103 specifies a hybrid peer-to-peer overlay network management protocol that runs on the interface between a hybrid peer and a hybrid overlay management server. This protocol provides management functionalities such as creation, removal, retrieval, and update, and describes the behaviour of protocols including the specifications of protocol messages.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Q.4103	2022-02-13	11	11.1002/1000/14924

Keywords

Hybrid peer-to-peer, overlay management, overlay network, protocol.

* 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/software copyrights, 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 appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Q.4103

Hybrid peer-to-peer communications: Overlay management protocol

1 Scope

This Recommendation describes the overlay management protocol for hybrid peer-to-peer communications as follows:

– protocol elements;

– protocol messages and its parameters;

– protocol behaviours including information flow.

2 References

The following ITU-T Recommendations and other references contain provisions that, 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 Q.4100] Recommendation ITU-T Q.4100 (2020), *Hybrid peer-to-peer communications: Functional architecture*.

[ITU-T Q.4101] Recommendation ITU-T Q.4101 (2021), *Hybrid peer-to-peer communications: Tree and data recovery procedures*.

[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 hybrid overlay network [ITU-T Q.4100]: A peer-to-peer overlay network in which participating peers exchange data using the pull and push method. The hybrid overlay network also provides a way to organize and maintain a tree-style path for pushing data to all peers without loops, as well as fetching data from other peers simultaneously.

3.1.2 hybrid peer [ITU-T Q.4100]: A peer capable of exchanging data using mesh-based and tree-based methods running over a hybrid overlay network.

3.1.3 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.

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

3.1.5 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.

3.1.6 reference point [b-ITU-T Y.2012]: A conceptual point at the conjunction of two non-overlapping functional entities that can be used to identify the type of information passing between these functional entities.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

FQDN Fully Qualified Domain Name

HOMP Hybrid Overlay Management Protocol

HON Hybrid Overlay Network

HP2P Hybrid Peer-to-Peer

IoT Internet of Things

P2P Peer-to-Peer

URI Uniform Resource Identifier

5 Conventions

Resource elements in clause 6.1 and fields of the message under clause 6.2 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", "DICT", "LIST" and "NUMBER" types are used to indicate string, Boolean, dictionary, list and number, respectively;

– An array of collective values are enclosed in brackets "[ ]" with values separated by commas ",";

– Selective options are separated by a vertical bar " | ".

6 Resource element types and message syntax of HOMP

This clause describes the message formats for the hybrid overlay management protocol (HOMP). For extensibility, HOMP is a representational state transfer (REST) architecture [b-Fielding] and messages are encoded in JavaScript object notation (JSON) [IETF RFC 7159].

6.1 Resource elements

This clause specifies the basic resource elements to be used for conveying information regarding the activities on the overlay network.

6.1.1 OVERLAY_INFO

The OVERLAY_INFO element include details about a hybrid overlay network (HON). This element contains overlay network attributes as shown in the following Table 6-1:

Table 6-1 – OVERLAY_INFO resource element

Keyword	Type	Description
overlay-id	STRING	*overlay-id* is an identifier for a specific HON.
title	STRING	*title* is a human readable description on the HON.
type	STRING	*type* indicates the type of the HON, and it should be one of 'core' or 'sub'.
sub-type	STRING	*sub-type* indicates the shape of the overlay topology, and it should be one of 'tree' or 'mesh'.
owner-id	STRING	*owner-id* indicates the peer identifier who created the HON.
expires	NUMBER	*expires* indicates the time that the HON will be removed in seconds.
status	OVERLAY_STATUS	*status* indicates the status of the HON such as the number of peers, peer list, and human readable status like 'active', 'terminated', etc.
description	STRING	*descriptions* provide more detailed information regarding the HON.
heartbeat-interval	NUMBER	*heartbeat-interval* indicates the interval for checking the aliveness of peers.
heartbeat-timeout	NUMBER	*heartbeat-timeout* indicates when a peer decides a corresponding peer is no longer alive.
auth	OVERLAY_AUTH	*auth* includes the authentication information to be used for access control to the HON.
cr-policy	CR_POLICY	*cr-policy* includes caching and recovery policies for the tree and data recovery policies of the HON.
trans-policy	TRANS_POLICY	*trans-policy* includes policy regarding data broadcasting floor control.

6.1.2 PEER_INFO

The PEER_INFO element includes details about peer information such as peer identifier, the network address and authentication information as shown in the following Table 6-2.

Table 6-2 – PEER_INFO resource element

Keyword	Type	Description
peer-id	STRING	*peer-id* includes the identifier of a hybrid peer.
Address	STRING	*Address* includes the network address that another peer is able to access.
auth	PEER_AUTH	*auth* includes authentication information needed for establishing a connection among peers.

6.1.3 OVERLAY_AUTH

The OVERLAY_AUTH element includes authentication details for managing the overlay network as shown in the following Table 6-3. This element is used for creating and manipulating the attributes of a hybrid overlay network.

Table 6-3 – OVERLAY_AUTH resource element

Keyword	Type	Description
type	STRING	*type* indicates whether it is public or not. If it is public, this value is 'open', if not, it is 'closed'.
admin_key	STRING	*admin_key* includes the encrypted key phrase for managing the hybrid overlay network. This is used only by the owner of this HON.
access_key	STRING	*access_key* includes the encrypted key phrase for accessing the hybrid overlay network. This value is valid when *type* is 'closed'.
peerlist	LIST[STRING]	*peerlist* includes the list of peer identifiers that have access grants. This value is valid when *type* is 'closed'. The listed peer can access the HON without an *access_key*.

6.1.4 OVERLAY_STATUS

The OVERLAY_STATUS element includes information to express the status of a hybrid overlay network as shown in following Table 6-4. It provides the number of peers within a particular HON, the list of peers and its status.

Table 6-4 – OVERLAY_STATUS resource element

Keyword	Type	Description
num_peers	NUMBER	*num_peers* indicates the number of peers of the HON.
peer_info_list	LIST[PEER_INFO]	*peer_info_list* includes the information of peers such as peer identifier, network address, etc.
status	STRING	*status* includes human readable information regarding the status of the HON, such as 'active', 'terminated' and so on.

6.1.5 PEER_STATUS

The PEER_STATUS element represents the status of a peer such as the number of primary / candidate path and the cost map that describes the routing path with other peers in detail as shown in the following Table 6-5.

Table 6-5 – PEER_STATUS resource element

Keyword	Type	Description
num_primary	NUMBER	*num_primary* indicates the number of current primary paths of the peer.
num_out_candidate	NUMBER	*num_out_candidate* indicates the number of current secured outgoing candidate paths of the peer.
num_in_candidate	NUMBER	*num_in_candidate* indicates the number of incoming candidate paths of the peer.
costmap	PEER_COSTMAP	*costmap* includes details on the routing path of the peer.

6.1.6 PEER_COSTMAP

The PEER_COSTMAP element includes details on the paths to adjacent peers of the peer as shown in the following Table 6-6.

Table 6-6 – PEER_COSTMAP resource element

Keyword	Type	Description
primary	LIST	*primary* includes the list of peer identifiers that are connected as a primary path.
outgoing_candidate	LIST	*outgoing_candidate* includes the list of peer identifiers connected as candidate paths initiated by this peer.
incoming_candidate	LIST	*incoming_candidate* includes the list of peer identifiers connected as candidate paths initiated by remote peers.

6.1.7 PEER_AUTH

The PEER_AUTH element includes authentication information for establishing a path with the peer as shown in the following Table 6-7. This is for preventing DoS attacks rather than authorization.

Table 6-7 – PEER_AUTH resource element

Keyword	Type	Description
password	STRING	*password* includes a key phrase for authenticating an incoming connection request. The key phrase consists of alpha-numeric characters only.

NOTE – The password is randomly generated whenever the hybrid peer instance is initiated

6.1.8 CR_POLICY

This resource element represents the caching and recovery policy of the overlay network, and all participating peer is required to obey this rule. The detailed parameters are described in Table 6-8.

Table 6-8 – CR_POLICY resource element

Keyword	Type	Description
mN_Cache	NUMBER	*mN_Cache* includes the number of messages to be cached. Each peer should make the circular queue bigger than this value. If this value is 0, the circular queue is not maintained. That is, the data recovery function is not provided.
mD_Cache	NUMBER	*mD_Cache* includes the minimum duration in seconds that each peer should maintain for each cached data.
recovery-by	STRING	*recovery-by* parameter is used when creating an overlay network requiring lost data recovery. The value will be one of "push" or "pull".

6.2 Messages syntaxes

This clause describes the messages for managing a hybrid overlay network to be used in the reference point R1 that is specified in [ITU-T Q.4100]. In the body of an HTTP request/response message, the following notation is used for representing the requirements on the specific parameter.

- m: mandatory
- o: optional
- c: conditional mandatory; requirement on the parameter depends on the context of the message usages.

Since the hybrid overlay management protocol runs on top of the HTTP [IETF RFC 7231], it obeys the semantics and syntaxes of the HTTP if not specifically described in this Recommendation.

6.2.1 HybridOverlayCreation

This *HybridOverlayCreation* primitive is used for creating a new hybrid overlay network. Table 6-9 describes the syntax of the request message.

Table 6-9 – Request message syntax for HybridOverlayCreation

<table>
<tr><td>HTTP method</td><td colspan="2">POST</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/homs</td></tr>
<tr><td>BODY</td><td>overlay_info</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>title</td><td>STRING</td><td>m</td></tr>
<tr><td>type</td><td>STRING</td><td>m</td></tr>
<tr><td>sub-type</td><td>STRING</td><td>m</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>o</td></tr>
<tr><td>description</td><td>STRING</td><td>m</td></tr>
<tr><td>heartbeat-interval</td><td>NUMBER</td><td>m</td></tr>
<tr><td>heartbeat-timeout</td><td>NUMBER</td><td>m</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>m</td></tr>
<tr><td>cr-policy</td><td>CR-POLICY</td><td>o</td></tr>
</table>
</td></tr>
</table>

– If the *expires* field is not specified, it assumes it as 86 400 seconds (one day).

– For preventing heartbeat storming, *heartbeat-interval* shall be bigger than 5 seconds, and *heartbeat-timeout* shall be bigger than *heartbeat-interval x 3* seconds.

– If *cr-policy* is not present, it means that the HON does not support data recovery.

– If *auth* is not present, the HON is open to the public.

Table 6-10 describes the syntax of the response for *HybridOverlayCreation*.

Table 6-10 – Response message syntax for HybridOverlayCreation

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>type</td><td>STRING</td><td>m</td></tr>
<tr><td>sub-type</td><td>STRING</td><td>m</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>m</td></tr>
<tr><td>heartbeat-interval</td><td>NUMBER</td><td>m</td></tr>
<tr><td>heartbeat-timeout</td><td>NUMBER</td><td>m</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>c</td></tr>
<tr><td>cr-policy</td><td>CR-POLICY</td><td>c</td></tr>
</table>
</td></tr>
</table>

c: If the parameter is present in the request message, this parameter must be included.

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

– *overlay-id* is an identifier that is created by HOMS for the creation request.

6.2.2 HybridOverlayQuery

This *HybridOverlayQuery* primitive is used for querying detailed information of a hybrid overlay network. Table 6-11 describes the syntax of the request.

Table 6-11 – Request message syntax for HybridOverlayQuery

<table>
<tr><td>HTTP method</td><td colspan="2">GET</td></tr>
<tr><td rowspan="2">HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/homs</td></tr>
<tr><td>URI params</td><td>
<table>
<tr><th colspan="3">URI params</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>o</td></tr>
<tr><td>title</td><td>STRING</td><td>o</td></tr>
<tr><td>description</td><td>STRING</td><td>o</td></tr>
</table>
</td></tr>
</table>

– If no params in the HTTP uniform resource identifier (URI), HOMS returns the list of HONs.

– If *description* parameter is present, the HOMS returns the list of HONs that contains the keyword within the description of HON.

– If *title* parameter is present, the HOMS returns the list of HONs that contains the keyword within the *title* of HON.

Table 6-12 describes the syntax of the response for *HybridOverlayQuery*.

Table 6-12 – Response message syntax for HybridOverlayQuery

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay[]</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>title</td><td>STRING</td><td>m</td></tr>
<tr><td>type</td><td>STRING</td><td>m</td></tr>
<tr><td>sub-type</td><td>STRING</td><td>m</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>STRING</td><td>m</td></tr>
<tr><td>status</td><td>OVERLAY_STATUS</td><td>m</td></tr>
<tr><td>description</td><td>STRING</td><td>o</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>m</td></tr>
<tr><td>cr-policy</td><td>CR_POLICY</td><td>c</td></tr>
<tr><td>trans_policy</td><td>TRANS_POLICY</td><td>o</td></tr>
</table>
</td></tr>
</table>

– If the *cr-policy* of the hybrid overlay network is present, it is included in the response message.

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

– In the HTTP body, HOMS returns the list of OVERLAY_INFO resource elements.

6.2.3 HybridOverlayModification

This *HybridOverlayModification* primitive is used for modifying the attributes of a hybrid overlay network. Table 6-13 describes the syntax of the request. By using this primitive, an owner of the HON is capable of modifying the title, description, expires and auth.

Table 6-13 – Request message syntax for HybridOverlayModification

<table>
<tr><td>HTTP method</td><td colspan="2">PUT</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/homs</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>title</td><td>STRING</td><td>o</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>o</td></tr>
<tr><td>description</td><td>STRING</td><td>o</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>m</td></tr>
<tr><td>trans_policy</td><td>TRANS_POLICY</td><td>o</td></tr>
</table>
</td></tr>
</table>

– In the *auth* parameter, *admin-key* is mandatory, and *access-key* and *peerlist* are optional.

Table 6-14 describes the syntax of the response for *HybridOverlayModification*.

Table 6-14 – Response message syntax for HybridOverlayModification

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>title</td><td>STRING</td><td>c</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>o</td></tr>
<tr><td>description</td><td>STRING</td><td>c</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>m</td></tr>
</table>
</td></tr>
</table>

– If *description* is present in the attributes of the hybrid overlay network, it is included in the response message.

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

– In the *auth* parameter, *admin-key, access-key and peerlist* are mandatory if the attributes are present for the HON.

6.2.4 HybridOverlayRemoval

This *HybridOverlayRemoval* primitive is used for removing a particular hybrid overlay network. Table 6-15 describes the syntax of the request.

Table 6-15 – Request message syntax for HybridOverlayRemoval

<table>
<tr><td>HTTP method</td><td colspan="2">DELETE</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/homs</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>owner-id</td><td>STRING</td><td>m</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>m</td></tr>
</table>
</td></tr>
</table>

– In the *auth* parameter, *admin-key* is mandatory.

Table 6-16 describes the syntax of the response for *HybridOverlayModification*.

Table 6-16 – Response message syntax for HybridOverlayRemoval

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
</table>
</td></tr>
</table>

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

6.2.5 HybridOverlayJoin

This *HybridOverlayJoin* primitive is used for joining an existing hybrid overlay network. Table 6-17 describes the syntax of the request.

Table 6-17 – Request message syntax for HybridOverlayJoin

<table>
<tr><td>HTTP method</td><td colspan="2">POST</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/peer</td></tr>
<tr><td rowspan="2">BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>type</td><td>STRING</td><td>m</td></tr>
<tr><td>sub-type</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>o</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>o</td></tr>
<tr><td>recovery</td><td>BOOLEAN</td><td>o</td></tr>
<tr><td>ticket-id</td><td>STRING</td><td>o</td></tr>
</table>
</td></tr>
<tr><td>peer</td><td>
<table>
<tr><th colspan="3">PEER_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>peer-id</td><td>STRING</td><td>m</td></tr>
<tr><td>address</td><td>STRING</td><td>m</td></tr>
<tr><td>auth</td><td>PEER_AUTH</td><td>m</td></tr>
</table>
</td></tr>
</table>

– If there is no refresh before the timer, specified in the *expires* parameter and the fields, the peer will be removed from the HON.

– When the requesting peer is already in the hybrid overlay network, *recovery* parameter is set to be *true*.

– When the requesting peer sends this message for refresh or recovery, *ticket-id* parameter must be present. That is, if the peer is already a member of the HON, this value must be embedded in the request message.

Table 6-18 describes the syntax of the response for *HybridOverlayJoin*.

Table 6-18 – Response message syntax for HybridOverlayJoin

<table>
<tr><td>HTTP response code</td><td colspan="3">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td colspan="2">
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>type</td><td>STRING</td><td>m</td></tr>
<tr><td>sub-type</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>m</td></tr>
<tr><td>status</td><td>OVERLAY_STATUS</td><td>m</td></tr>
<tr><td>trans_policy</td><td>TRANS_POLICY</td><td>o</td></tr>
</table>
</td></tr>
</table>

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

– *overlay-id* is an identifier that is created by HOMS for the creation request.

6.2.6 HybridOverlayReport

This *HybridOverlayReport* primitive is used for reporting the status of a peer. Table 6-19 describes the syntax of the request.

Table 6-19 – Request message syntax for HybridOverlayReport

<table>
<tr><td>HTTP method</td><td colspan="3">POST</td></tr>
<tr><td>HTTP URI</td><td colspan="3">http://[HOMS-FQDN]:[port]/peer/report</td></tr>
<tr><td rowspan="3">BODY</td><td>overlay</td><td colspan="2">
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>string</td><td>m</td></tr>
</table>
</td></tr>
<tr><td>peer</td><td colspan="2">
<table>
<tr><th colspan="3">PEER_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>peer-id</td><td>string</td><td>m</td></tr>
<tr><td>auth</td><td>PEER_AUTH</td><td>m</td></tr>
</table>
</td></tr>
<tr><td>status</td><td colspan="2">
<table>
<tr><th colspan="3">PEER_STATUS</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>status</td><td>PEER_STATUS</td><td>m</td></tr>
</table>
</td></tr>
</table>

Table 6-20 describes the syntax of the response for *HybridOverlayReport.*

Table 6-20 – Response message syntax for HybridOverlayReport

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>o</td></tr>
</table>
</td></tr>
</table>

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses appropriate HTTP response code if it fails.

– If *overlay-id* is not included, *overlay* is also emitted. That is, the HTTP body of the response may be empty.

6.2.7 HybridOverlayRefresh

This *HybridOverlayRefresh* primitive is used for continuous stay in the overlay network by a peer. A peer needs to send this message periodically to prevent it from being removed from the peer list of a specific overlay network due to timeout. Table 6-21 describes the syntax of the request.

Table 6-21 – Request message syntax for HybridOverlayRefresh

<table>
<tr><td>HTTP method</td><td colspan="2">PUT</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/peer</td></tr>
<tr><td rowspan="2">BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>o</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>o</td></tr>
</table>
</td></tr>
<tr><td>peer</td><td>
<table>
<tr><th colspan="3">PEER_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>peer-id</td><td>STRING</td><td>m</td></tr>
<tr><td>address</td><td>STRING</td><td>m</td></tr>
<tr><td>auth</td><td>PEER_AUTH</td><td>m</td></tr>
</table>
</td></tr>
</table>

Table 6-22 describes the syntax of the response for *HybridOverlayRefresh.*

Table 6-22 – Response message syntax for HybridOverlayRefresh

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td rowspan="2">BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>expires</td><td>NUMBER</td><td>m</td></tr>
</table>
</td></tr>
<tr><td>peer</td><td>
<table>
<tr><th colspan="3">PEER_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>peer-id</td><td>STRING</td><td>m</td></tr>
</table>
</td></tr>
</table>

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

6.2.8 HybridOverlayLeave

This *HybridOverlayLeave* primitive is used for leaving an overlay network explicitly for graceful leaving. If a peer leaves without sending this message, HOMS will remove it after a timeout or another peer can detect the disconnection of the primary path with the leaving peer. If the peer detects the event, it initiates the tree recovery procedures specified in [ITU-T Q.4101]. On successful recovery of the network, each peer sends a *HybridOverlayReport* message to HOMS.

Table 6-23 describes the syntax of the request.

Table 6-23 – Request message syntax for HybridOverlayLeave

<table>
<tr><td>HTTP method</td><td colspan="2">DELETE</td></tr>
<tr><td>HTTP URI</td><td colspan="2">http://[HOMS-FQDN]:[port]/peer</td></tr>
<tr><td rowspan="2">BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
<tr><td>auth</td><td>OVERLAY_AUTH</td><td>c</td></tr>
</table>
</td></tr>
<tr><td>peer</td><td>
<table>
<tr><th colspan="3">PEER_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>peer-id</td><td>STRING</td><td>m</td></tr>
<tr><td>auth</td><td>PEER_AUTH</td><td>m</td></tr>
</table>
</td></tr>
</table>

– If the overlay network is public, *auth* field is not included.

Table 6-24 describes the syntax of the response for *HybridOverlayLeave.*

Table 6-24 – Response message syntax for HybridOverlayLeave

<table>
<tr><td>HTTP response code</td><td colspan="2">rsp_code</td></tr>
<tr><td>BODY</td><td>overlay</td><td>
<table>
<tr><th colspan="3">OVERLAY_INFO</th></tr>
<tr><th>keyword</th><th>type</th><th>m/o</th></tr>
<tr><td>overlay-id</td><td>STRING</td><td>m</td></tr>
</table>
</td></tr>
</table>

– *rsp_code* indicates the result of the request. If successful, this is 200, and it uses the appropriate HTTP response code if it fails.

7 Protocol operations and information flows

The hybrid overlay management protocol (HOMP) consists of two parts; hybrid overlay network management and hybrid overlay peer management.

7.1 Hybrid overlay network management

This clause describes the functionalities for managing a hybrid overlay network such as creation, query, modification and deletion.

7.1.1 Creation of hybrid overlay network

As shown in Figure 7-1, a peer sends the *HybridOverlayCreation* request message to HOMS to create a new hybrid overlay network. The peer keeps ownership of the network, and it has the privilege to update or delete it.

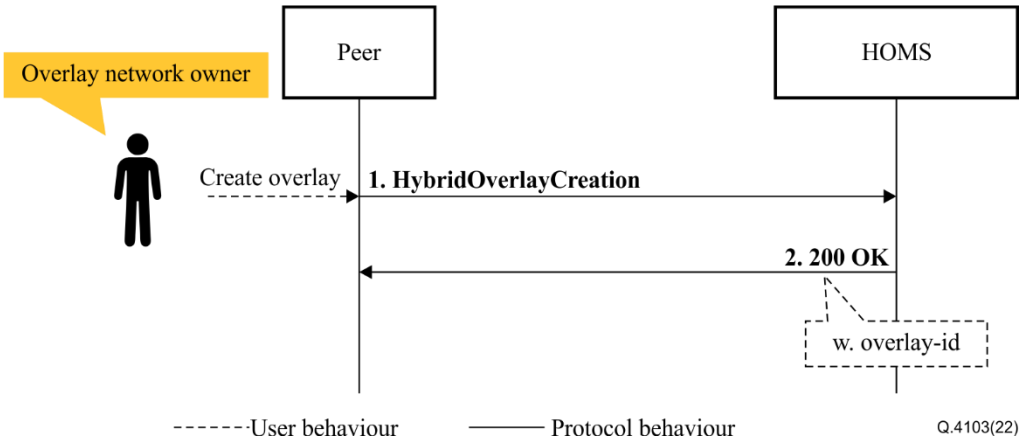

Figure 7-1 – Information flow for hybrid overlay network creation

1 A peer sends a *HybridOverlayCreation* request message to create a HON and uses a parameter of message defined in clause 6 to specify the properties of an overlay network.

2 The HOMS receiving the *HybridOverlayCreation* message creates the requested HON, and when successfully created, transmits the network attribute information including the newly issued overlay-id to the peer through a 200 OK response.

7.1.2 Query of hybrid overlay network

When a user wants to get the list of the overlay network with details, the user sends *HybridOverlayQuery* messages to HOMS. This clause describes the operational procedures for querying the details on a specific hybrid overlay network that is already created by any peer as specified in clause 7.1.1, as shown in Figure 7-2.

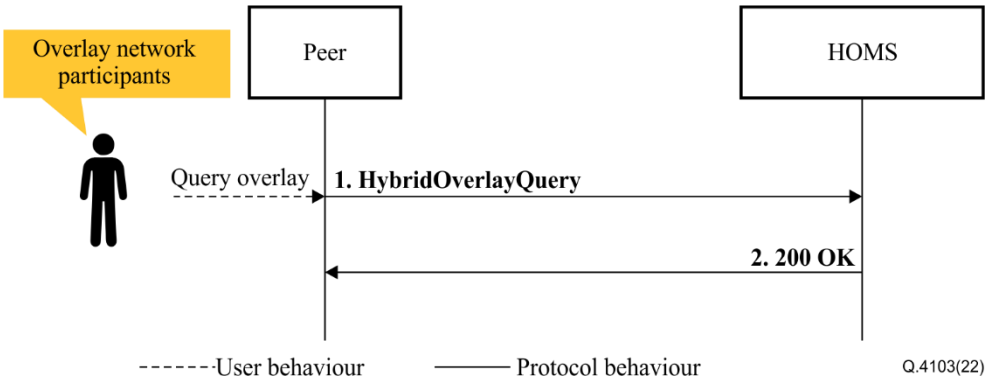

Figure 7-2 – Information flow for hybrid overlay network query

1 A peer sends a *HybridOverlayQuery* request message to HOMS, and this request message can contain keywords to be queried such as overlay-id, title, description, etc.

2 The HOMS receiving the *HybridOverlayQuery* message returns the requested information within the body of the 200 OK response. If there is no keyword in the URI parameter of the request message, HOMS returns the list of the current available overlay networks.

7.1.3 HybridOverlayModification

This clause describes the operational procedures for modifying the details on a specific hybrid overlay network. Figure 7-3 shows the information flow for the modification of a hybrid overlay network.

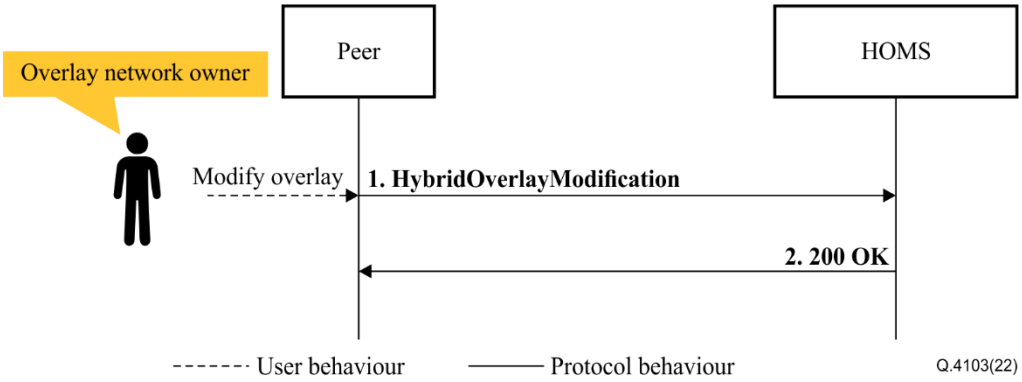

Figure 7-3 – Information flow for hybrid overlay network modification

1 A peer sends a *HybridOverlayModification* request message to HOMS to modify attributes of a specific overlay network. The overlay network owner is able to modify the *description*, *title*, *expires* and *auth*.

NOTE – The updated information is reflected immediately by HOMS but does not take effects from already joined peers. On periodic refreshing by a peer, peers will receive the latest information.

2 The HOMS receiving the *HybridOverlayModification* message returns details of the overlay network to check the modification result by the peer.

7.1.4 HybridOverlayRemoval

This clause describes the operational procedures for removing a specific hybrid overlay network by an owner. Figure 7-4 shows the information flow for removing a hybrid overlay network by an owner.

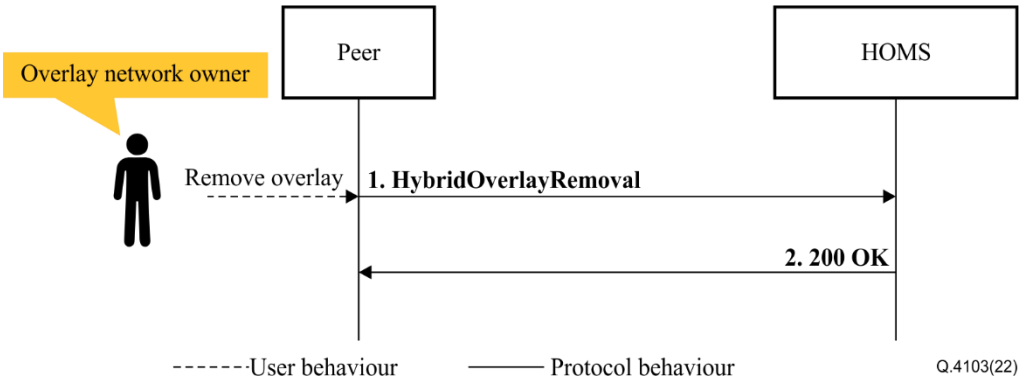

Figure 7-4 – Information flow for hybrid overlay network removal

1 A peer sends a *HybridOverlayRemoval* request message to HOMS to remove a specific overlay network explicitly.

2 The HOMS receiving the *HybridOverlayRemoval* message returns the 200 OK messages.

NOTE 1 – This operation just removes information in HOMS, and HOMS does not notify the event to the peers of the overlay network. When a peer refreshes its status, it will find out that the overlay network is no longer valid. However, the implementation or service is dependent on whether the HON is leaving the network or not.

NOTE 2 – When the lifetime of the overlay network is expired, HOMS removes the overlay network information silently. If the owner wants to extend the lifetime, it updates by using the *HybridOverlayModification* message.

7.2 Hybrid overlay peer management

This clause describes the procedures related to a peer's activity such as join, leave, report and refresh.

7.2.1 HybridOverlayJoin

This clause describes the operational procedures for joining an overlay network by a peer. It is assumed that the peer has an overlay-id to join an overlay network, and it can be acquired by the use of the HybridOverlayQuery message specified in clause 7.1.2 of this Recommendation. Figure 7-5 shows the information flow for joining into an existing hybrid overlay network.

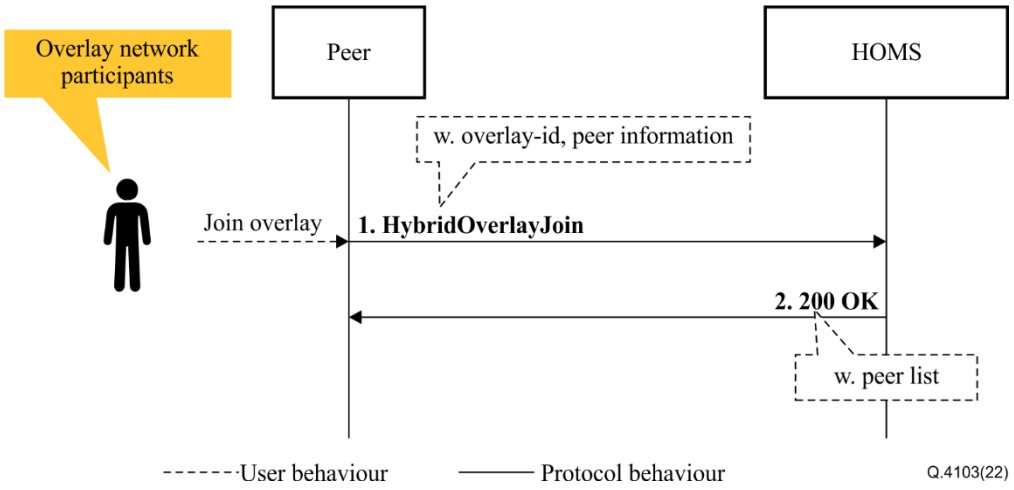

Figure 7-5 – Information flow for hybrid overlay network join

1 A peer sends a *HybridOverlayJoin* request message to HOMS to join a specific overlay network. In the request message, it includes the overlay-id as well as the peer information as specified in clause 6.2.5 of this Recommendation.

2 The HOMS receiving the *HybridOverlayRemoval* message returns the 200 OK message with a peer list that contains the entry peer list as specified in clause 6.2.5.

7.2.2 HybridOverlayReport

This clause describes the operational procedures for reporting the status of a peer. When a peer detects an event to be reported to HOMS, it sends a *HybridOverlayReport* message. The reported information is used to optimize and analyse the overlay network. HOMS can figure out the structure of CoreTree by using aggregated reports from many peers simultaneously. The types of events include the following but are not limited:

– changes of primary/candidate path;

– attachment of a new application on the peer;

– etc.

Figure 7-6 shows information flows for reporting peer's status to HOMS.

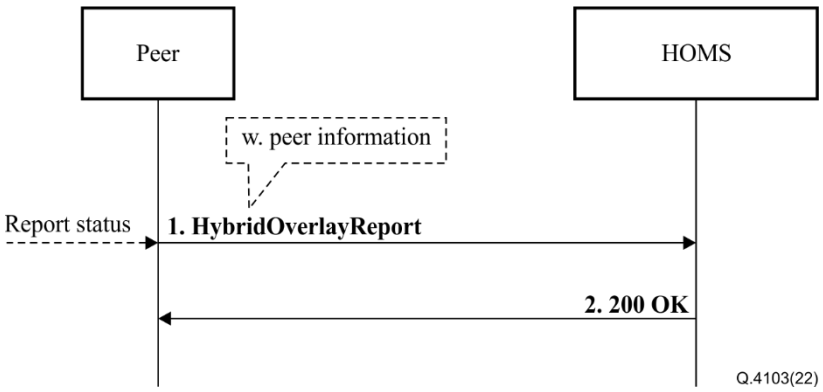

Figure 7-6 – Information flow for hybrid overlay network report

1 A peer sends a *HybridOverlayReport* request message to HOMS to report the status of the peer.

2 The HOMS receiving the *HybridOverlayReport* message returns a 200 OK message with overlay information as specified in clause 6.2.6.

7.2.3 HybridOverlayRefresh

This clause describes the operational procedures for a peer's prolong joining to a specific HON and updating the application attachment to the peer. When a peer joins a HON, there is a lifetime for the join that is specified in the expire parameter of the response message. When the lifetime expires, the peer is excluded from the overlay network. In order to keep the join, the peer needs to send a *HybridOverlayRefresh* message periodically before the expiration. In addition, it needs to also report to HOMS on any attachment of a new data prosumer application such as Internet of things (IoT), multimedia, blockchain, etc.

Figure 7-7 shows the information flow for refreshing the peer's joining to a specific HON.

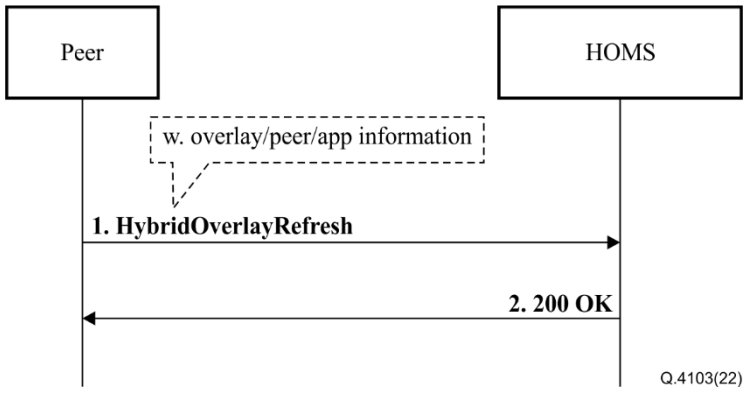

Figure 7-7 – Information flow for hybrid overlay network refresh

1 A peer sends a *HybridOverlayRefresh* request message to HOMS to refresh the status of the peer.

2 The HOMS receiving the *HybridOverlayRefresh* message returns a 200 OK message with the overlay information as specified in clause 6.2.7. On sending the response, the HOMS extends the lifetime of the peer's joining and embeds the updated lifetime into the *expires* parameter in the overlay information.

7.2.4 HybridOverlayLeave

This clause describes the operational procedures for leaving a HON by a peer explicitly. These procedures are performed after the peer release peer relationship with other peers. On receiving this request, HOMS remove the peer from the overlay network immediately. If a peer leaves without these

procedures, it will remain in the overlay network for the time being before its expiration, and HOMS will provide the peer's information to another peer's *HybridOverlayJoin* request. However, other peers do not make a connection with this peer and it causes no problem. Furthermore, peers connected with the ungracefully leaving peer detects the disconnection with the peer and the HON gets recovered immediately as specified in [ITU-T Q.4101].

Figure 7-8 shows the information flow for leaving a specific HON by a peer.

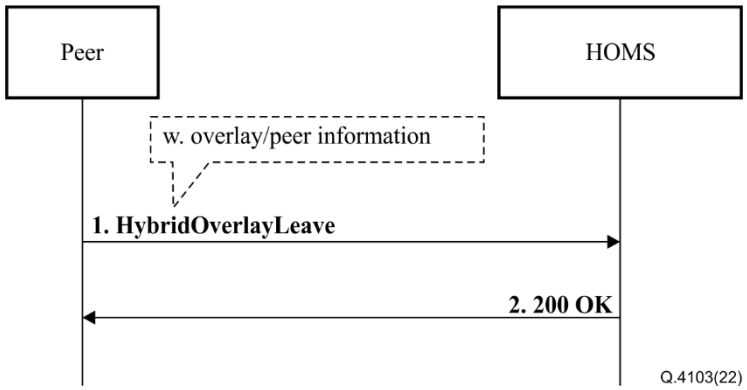

Figure 7-8 – Information flow for hybrid overlay network leave

1 A peer sends a *HybridOverlayLeave* request message to HOMS to leave a specific HON. On sending this request, the request message includes the PEER_AUTH elements for authenticating the peer by HOMS.

2 The HOMS receiving the *HybridOverlayLeave* message returns the 200 OK message, and removes the peer from the HON.

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.2012] Recommendation ITU-T Y.2012 (2010), *Functional requirements and architecture of next generation networks*.

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

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

<https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

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, 2022