Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 10 October 2026

Z. Luo, Ed.
H. Yan
CMCC

8 April 2026

Available Session Recovery Protocol
draft-cmcc-asrp-06

Abstract

This document describes an experimental protocol named the Available Session Recovery Protocol (ASRP). The protocol is designed to optimize high-availability network cluster architectures, providing a superior high-availability solution for clusters offering stateful network services such as load balancing and Network Address Translation (NAT [RFC4787]). ASRP defines the procedures for session backup and recovery, as well as the message formats used during these interactions, enabling efficient and streamlined session state management.

In contrast to traditional high-availability techniques that back up session state within the cluster itself, the core innovation of ASRP lies in its distributed backup of state information to the client or server side. This approach offers multiple advantages: theoretically unlimited elastic scaling capacity; support for rapid recovery from multi-point failures; reduction of resource redundancy through the elimination of centralized backup nodes; and significant simplification of cluster implementation complexity.

The ASRP protocol provides a standardized method for constructing elastic service clusters, facilitating broader participation from software and hardware developers in building elastic cloud network service clusters.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

Traditional high-availability network clusters based on a master-backup architecture rely on session state synchronization between the master and backup nodes. While functionally complete, this architecture faces challenges in the cloud era, such as insufficient flexibility for elastic scaling, resource redundancy, and high implementation complexity. To address these challenges, the industry has proposed the Elastic Stateful Cluster.

An Elastic Stateful Cluster is a high-availability network service cluster composed of multiple cooperative nodes. The number of nodes within the cluster can be elastically scaled, enabling it to provide stateful network services such as load balancing (SLB) and Network Address Translation (NAT). To achieve elastic scaling, conventional Elastic Stateful Clusters adopt a Fast/Slow Path design philosophy, separating session management from packet forwarding. This allows the fast path node layer to achieve good elastic scaling capabilities.

1.1. Conventional Elastic Stateful Cluster

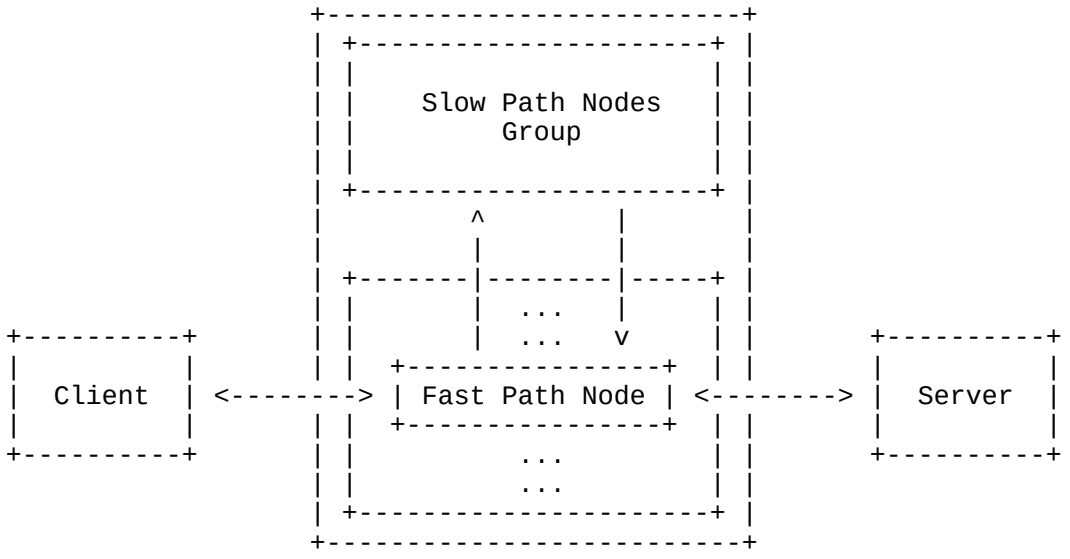

Figure 1: Fast/Slow Path Elastic Stateful Cluster

The slow path nodes are responsible for session creation and synchronization, while the fast path nodes are responsible for rapid packet forwarding. The drawback of this Elastic Stateful Cluster architecture is the weak elastic scaling capability of the slow path nodes. Implementing session synchronization among slow path nodes is complex. A typical implementation reference is the AWS Hyperplane NFV platform.

1.2. ASRP Elastic Stateful Cluster

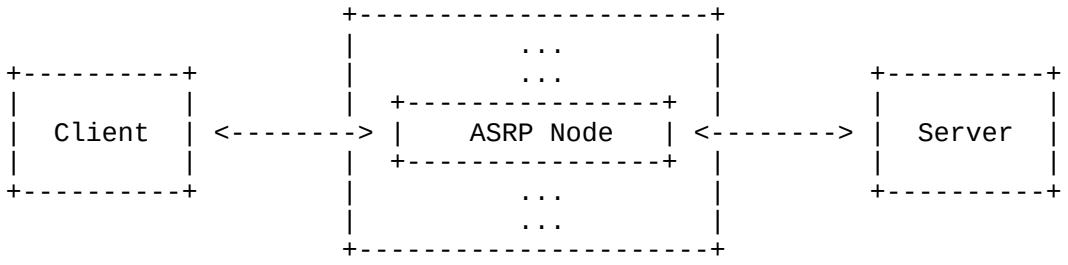

Figure 2: ASRP Elastic Stateful Cluster

The Available Session Recovery Protocol (ASRP) proposes an innovative high-availability solution, aiming to provide a standardized method for constructing elastic service clusters. This facilitates broader participation from software and hardware developers in building elastic cloud network service clusters. Its core idea is to innovatively distribute session state information to the client or server. The lifecycle of the backup state is synchronized with the real session, eliminating the need for independent keepalive and timeout mechanisms. This design ensures the timeliness and availability of the backup information.

ASRP defines corresponding session backup and recovery mechanisms. The protocol allows protocol messages to be transmitted together with the original service data packets, thereby reducing control overhead for state synchronization. In an elastic stateful cluster built on ASRP, network nodes possess atomic and mutually independent properties. There is no need for communication between nodes, nor is session synchronization required within the cluster. This fundamental design provides theoretically unlimited scaling capability and supports rapid recovery from multi-point failures.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Protocol Overview

3.1. Two Operational Modes

For the ASRP protocol to function correctly, two prerequisites must be met. First, all network nodes within the cluster MUST run service software supporting the ASRP protocol. Second, the server or client responsible for backing up sessions MUST deploy a kernel module or an eBPF module that supports ASRP. Depending on whether this module is deployed on the server or the client, the protocol operates in one of two corresponding modes: Passive (PSV) Mode and Active (ACT) Mode.

3.1.1. PSV Mode

In PSV mode, the network node is typically located within the same trusted network domain as the server (e.g., inside a data center). Its typical service is load balancing.

3.1.2. ACT Mode

In ACT mode, the network node is typically located within the same trusted network domain as the client (e.g., an enterprise intranet). Its typical service is Source Network Address Translation (SNAT).

3.2. Two Routing Behaviors

3.2.1. Symmetric Routing

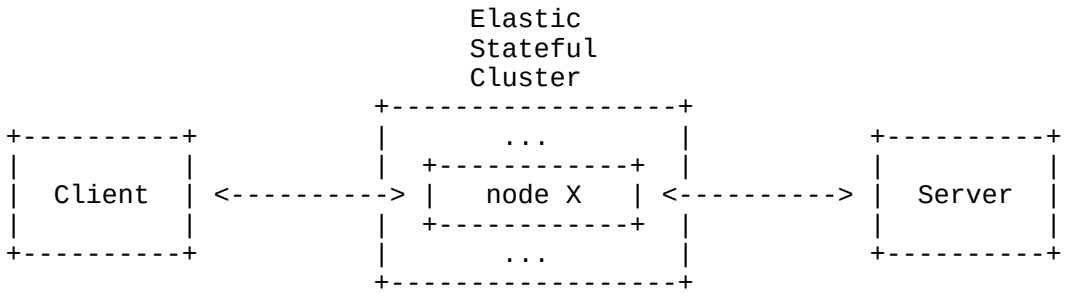

Figure 3: Symmetric Routing

Symmetric routing refers to the path mode where bidirectional traffic of the same session between a client and a server is always routed to the same node within the cluster.

3.2.2. Asymmetric Routing

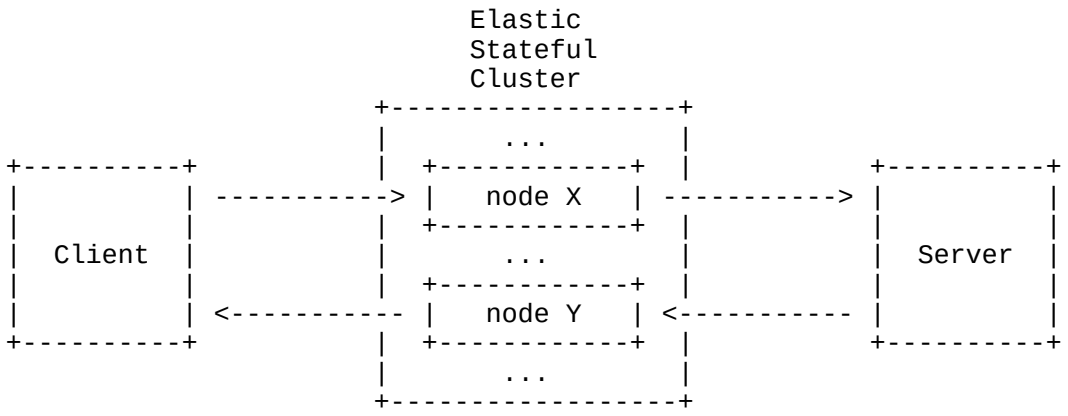

Figure 4: Asymmetric Routing

Asymmetric routing refers to the scenario where bidirectional traffic of the same session may be routed (e.g., by mechanisms such as ECMP [RFC2991], [RFC2992]) to different nodes within a cluster. In cloud networking environments, asymmetric routing is a common phenomenon, which imposes higher demands on the implementation of elastic stateful clusters.

3.3. Protocol Message

ASRP achieves distributed backup and recovery of session state information by exchanging specific protocol messages among the client, server, and network nodes (such as load balancers or NAT devices). In a load-balancing scenario, session state is distributed and backed up to individual servers; in a Source Network Address Translation (SNAT) scenario, session state is distributed and backed up to individual clients.

ASRP defines the following protocol messages: New Session message (NS), New-session Acknowledge message (NA), Query Session message (QS), Recover Session message (RS), Recovery no-session message (RX), Hello Session message (HS) and Push Session message (PS).

3.3.1. NS Message

Generated by the network node, it is used to send session state information to a designated client (in ACT mode) or server (in PSV mode) for backup when creating a new session.

3.3.2. NA Message

Generated by the server holding backup as response to a NS message in PSV mode, indicating that the NS message is received by the server.

3.3.3. QS Message

Generated by the network node, it is used to query the client or server for backup session state information when a received packet cannot match any local session and a session cannot be directly created. For TCP SYN packets, if no local session matches, a session can be created directly without querying the state.

3.3.4. RS Message

Generated by the client or server holding the backup as a response to a QS message, it contains the state information required to recover the session. The network node parses the RS message and reconstructs or marks the local session, thereby achieving failure recovery.

3.3.5. RX Message

Generated by the client or server holding the backup as a response to a QS message, indicating that the session queried by the QS message was not found. Except for the Msg Type field, the RX message is identical to the corresponding QS message.

3.3.6. HS Message

Generated by the client, it is used in ACT mode to announce to the network node its capability to support the ASRP protocol and to trigger the network node to return an NS message to complete session backup.

3.3.7. PS Message

Generated by the server, it is used in PSV mode to push session state information to the network node. In the case of asymmetric routing, the network node utilizes the PS message to create/update sessions for fast packet forwarding.

3.4. Transmission Modes and Signature

ASRP messages can be transmitted in three modes.

3.4.1. Inline mode

The ASRP message is inserted at the beginning of the payload of the forwarded packet. The ASRP message and the forwarded packet are transmitted together. When the total packet length does not exceed the MTU, NS, HS, and PS messages prefer this transmission mode.

3.4.2. Standalone mode

The ASRP message is encapsulated into a separate new packet that uses the same 5-tuples (source IP, destination IP, source port, destination port, and protocol) as the forwarded packet. This ensures that both the ASRP message and the forwarded packet are delivered to the same network node for processing. To avoid exceeding the MTU, NS, HS, and PS messages may use this transmission mode.

3.4.3. Bundled mode

The ASRP message is encapsulated using IP/UDP [RFC0768], with IP addresses configured to ensure mutual reachability and a fixed destination port ASRP_PORT (e.g., 51200). When the total packet length does not exceed the MTU, the original IP packet to be forwarded may optionally be appended after the ASRP message, allowing them to be transmitted together. QS, RS, RX and NA messages typically use this transmission mode; additionally, NS messages also use this mode when an NA response is expected.

3.4.4. ASRP Signature

In inline/standalone mode, an ASRP Signature is used to indicate that a packet contains an ASRP message.

Similar to the Proxy Protocol, a 12-byte ASRP Signature is used:

0x0D, 0x0A, 0x0D, 0x0A, 0x00, 0x0D, 0x0A, 0x41, 0x53, 0x52, 0x50, 0xF1.

The ASRP Signature is inserted in front of the ASRP message. In bundled mode, UDP destination port is ASRP-PORT, it does not require the ASRP Signature.

3.5. Session Creation/Recovery Scenarios

This section elaborates on, through a series of typical scenarios, how the ASRP protocol achieves session backup and recovery via message interaction in the event of network node failures under different operational modes. Each scenario details the involved protocol message flows and the processing steps of each entity.

3.5.1. PSV-Scenario-1

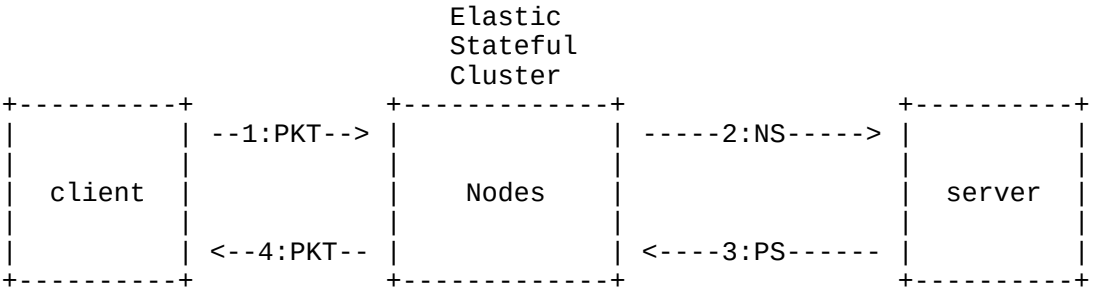

Figure 5: Direct Session Creation in PSV Mdoe

This scenario describes that, in PSV mode, a network node receives an explicit first packet (i.e., a packet whose characteristics explicitly indicate the start of a session) and directly creates a session flow. Common examples of explicit first packets include TCP SYN [RFC9293] and DNS [RFC1034] [RFC1035] query, among others.

The processing flow is as follows:

1. Session Creation: Upon receiving a packet from the client (e.g., TCP SYN), the network node first creates a new session and then sends an NS message to the selected server.

2. Server Response: Upon receiving the NS message, the server stores the session state information contained in the NS message and associates it with its local session. In the case of asymmetric routing, when sending its first response packet, the server sends a PS message to the network node.

3. Session Recovery: In the case of asymmetric routing, the network node, upon receiving the PS message, restores the session and subsequently forwards packets according to that session.

The session state information backed up by the server is released upon local session termination, without requiring any additional teardown message.

3.5.2. PSV-Scenario-2

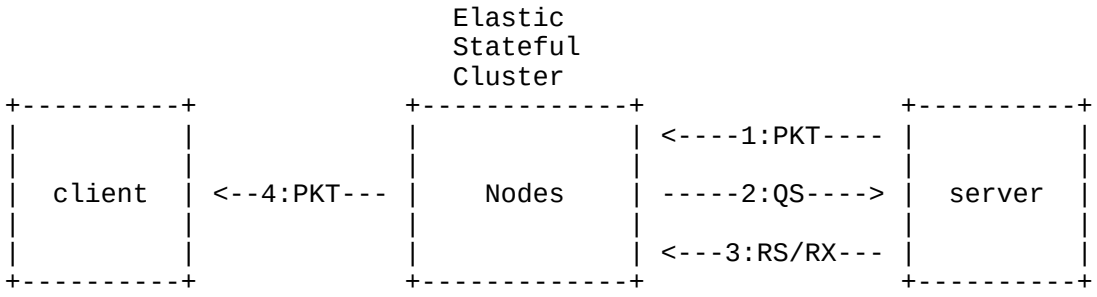

Figure 6: Session Recovery for Server in PSV Mode

This scenario describes the session recovery flow triggered by a server packet.

The processing flow is as follows:

1. Session Query: Upon receiving a packet from the server, the network node searches its local session table. If no matching session is found, the node SHOULD first buffer the packet for forwarding, then sends a QS message back to the server.

2. Server Response: After receiving the QS message, the server, based on the content of the QS message, looks up the locally stored backup session state information and sends an RS/RX message back to the network node.

3. Session Recovery: Upon receiving an RS message, the network node creates a new local session and forwards packets accordingly. Upon receiving an RX message, the node SHOULD discard the pending packets.

3.5.3. PSV-Scenario-3

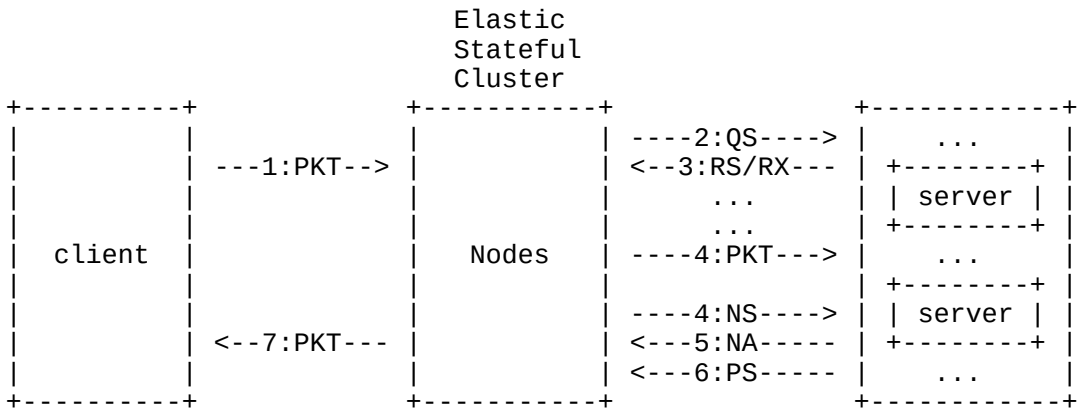

Figure 7: Session Creation/Recovery for Client in PSV Mode

This scenario describes the session creation/recovery flow triggered by a client packet.

The processing flow is as follows:

1. Query Local Session: Upon receiving a packet from the client, if no matching session is found, the network node first obtains a list of candidate servers (possibly multiple) for querying.

2. Query Backup Session: The network node sends QS messages to each candidate server to query for backed-up sessions. Each server replies with an RS or RX message indicating the query result.

3. Process Query Results: If a session is found, the network node restores the session according to the RS message and forwards the packet. Otherwise, for TCP packets: drop the packet. For UDP packets: proceed to create a new local session and send an NS message to the selected server.

4. Server Creates New Session: Upon receiving an NS message, the server stores the session state information and associates it with its local session. It immediately replies with a pure NA packet as an acknowledgment.

5. Session Recovery: In an asymmetric routing environment, when sending its first response packet, the server prioritizes sending a PS message to restore the session at the network node before forwarding the response packet.

In this scenario, obtaining the list of candidate servers is a key challenge. Two solutions are proposed:

1. The network node employs a deterministic server selection algorithm-such as consistent hashing or history-aware consistent hashing-to quickly map incoming packets to backend servers.

2. Enhance the client so that client packets carry backend server information, allowing the network node to directly extract the target server from the client packet.

3.5.4. ACT-Scenario-1

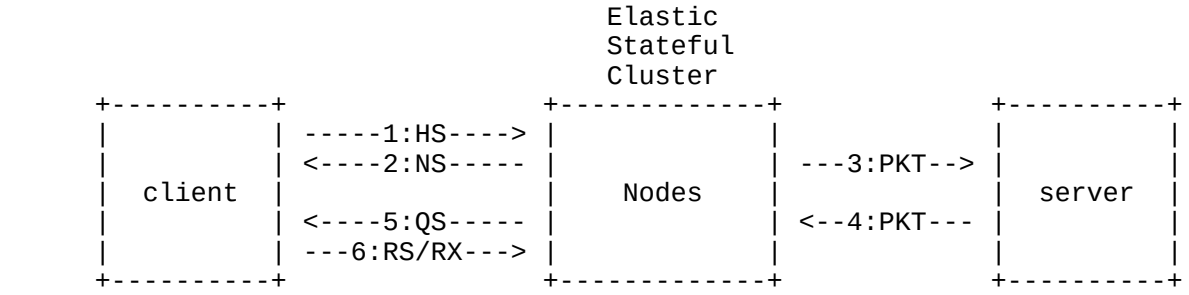

Figure 8: Session Creation/Recovery in ACT Mode

This scenario describes session creation at a network node and server-initiated session restoration.

The processing flow is as follows:

1. Session backup: When sending a packet without receiving NS message, the client sends an HS message to the network node to request session backup (the client should rate-limit HS). Upon receiving HS, the node replies with an NS message for session backup.

2. Session lookup: For packets from a server with no matching session, the network node identifies the target client and sends it a QS message.

3. Session Recovery: Upon receiving QS, the client sends an RS/RX message to the network node to restore the session.

In step 2, identifying the target client is challenging. Two solutions are proposed:

1. Use static mapping (e.g., map destination port to client). For SNAT, client IP addresses can be statically mapped to distinct port ranges.

2. Enhance the server to embed client information in its packets, allowing the network node to extract the client address directly.

3.5.5. ACT-Scenario-2

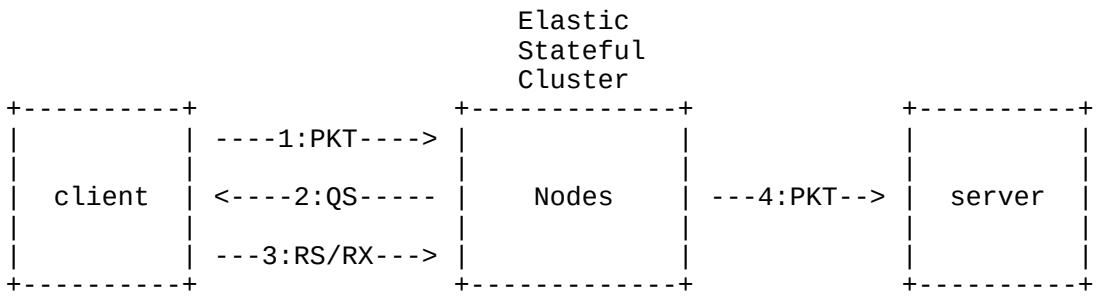

Figure 9: Session Recovery for Client in ACT Mode

This scenario describes the client-packet-triggered session recovery.

The processing flow is as follows:

1. Session Query: Upon receiving a packet from a client with no local session and without an HS message, the network node sends a QS message to the client.

2. Session Recovery: The client responds with an RS/RX message; the RS message enables session restoration.

4. Protocol Details

4.1. Message Format

An ASRP message consists of nine fields in total. The message header has a fixed length and is composed of the first six fields. The message body has variable length and comprises the remaining three fields. The fields are defined as follows:

1. Version: 1 octet, the protocol version.
2. MsgType: 1 octet, the message type.
3. Flags: 1 octet, message flags.
    - F_MSG: Pure message (sent independently);
    - F_ACT: ACT mode in use;
    - F_PEER: First ST is the server-side ST.
4. DataType: 1 octet, session tuple type.
5. Length: 2 octets, total ASRP message length in octets.
6. Reserved: 1 octet, reserved for future use.
7. Protocol: 1 octet, transport-layer protocol(e.g., TCP, UDP).
8. Session-Tuple(ST): source and destination addresses and ports.
    The IP address type is IPv4/IPv6.
    There are 6 types of ST, as follows:
    - ST4: IPv4-only tuple;
    - ST6: IPv6-only tuple;
    - ST44/ST66: Pairs of ST4 or ST6;
    - ST46/ST64: Mixed IPv4/IPv6 tuples.
9. Session-Data(SD): opaque session state information.

The values and semantics of the six header fields (Version through Reserved) are specified in the following table.

Field	Value	Name	Description
Version	0	Ver0	ASRP Version
MsgType	0	NS	NS Message
	1	NA	NA Message
	2	QS	QS Message
	3	RS	RS Message
	4	RX	RX Message
	5	HS	HS Message
	6	PS	PS Message
DataType	0	NULL	Message body contains no ST
	1	ST44	Message body contains ST44
	2	ST66	Message body contains ST66
	3	ST46	Message body contains ST46
	4	ST64	Message body contains ST64
Flags	0x1	F-MSG	Pure message
	0x2	F-ACT	ACT mode
	0x4	F-PEER	Peer Session-Tuple
Length	len	LEN	Message length in octets
Reserved	0	RSRV	Reserved

Figure 10: ASRP Message Header

ST4(length 12) Format:

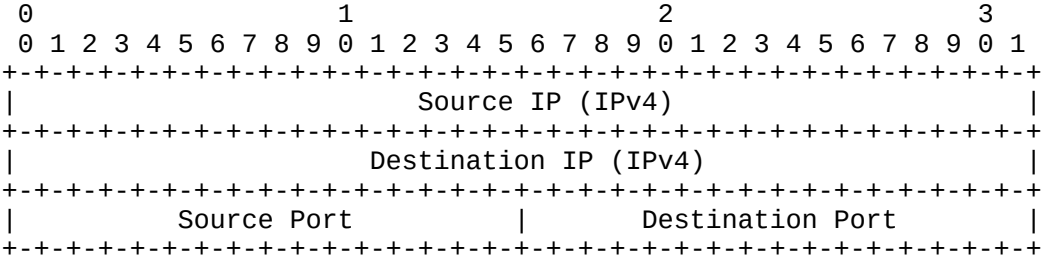

Figure 11: IPv4 Session Tuple Format

ST6(length 36) Format: Same structure as ST4, but with IPv6
ST44(length 24) Format: ST4 pair
ST66(length 72) Format: ST6 pair
ST46(length 48) Format: Mixed ST pair(ST4->ST6 sequence)
ST64(length 48) Format: Mixed ST pair(ST6->ST4 sequence)

4.1.1. NS Message Format

The NS message is used by the network node to back up session state information to the client or server. The NS message contains two Session-Tuples.

NS Message Format:

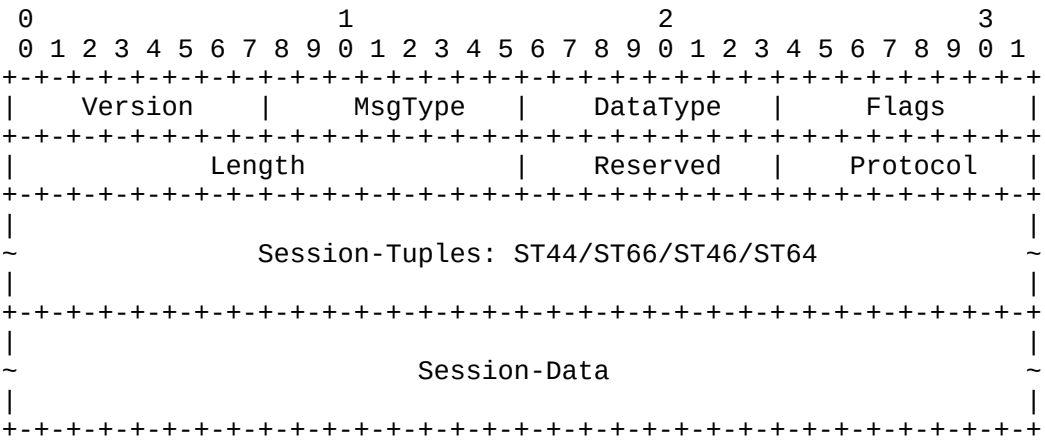

Figure 12: ASRP NS Message Format

The NS message contains two Session-Tuples, representing the connection between the network node and the client, and the connection between the network node and the server, respectively.

4.1.2. NA Message Format

The NA message is used only in PSV mode to inform the network node that the server has received the NS message.

RS Message Format: The structure of message is the same as that of the NS message.

4.1.3. QS Message Format

The QS message is used by the network node to query backup session state information.

QS Message Format:

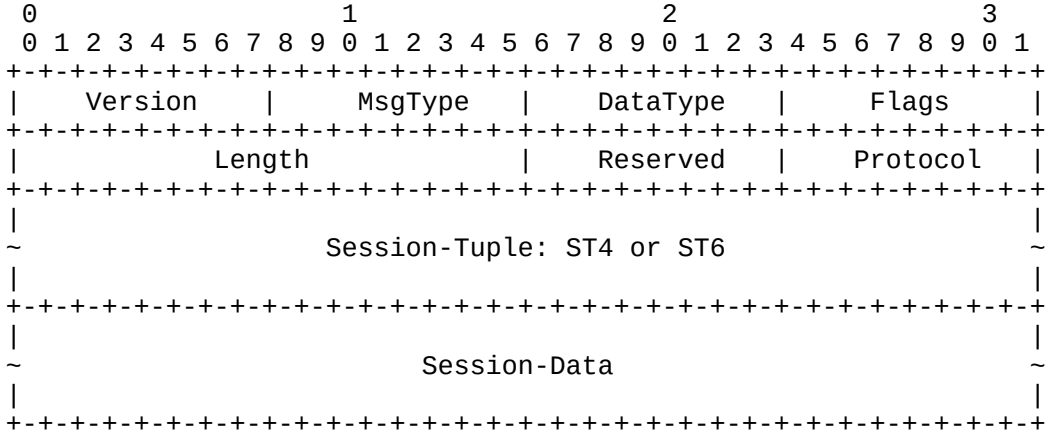

Figure 13: ASRP QS Message Format

4.1.4. RS Message Format

The RS message is used to recover a network node’s session.

RS Message Format: The structure of message is the same as that of the NS message.

4.1.5. RX Message Format

The RX message indicates that the session queried by a prior QS message is not present in the responder’s session table.

RX Message Format: The structure of messages is the same as taht of the QS message.

4.1.6. HS Message Format

The HS message is generated by the client to announce to the network node that it requires an NS message to back up session state information.

HS Message Format:

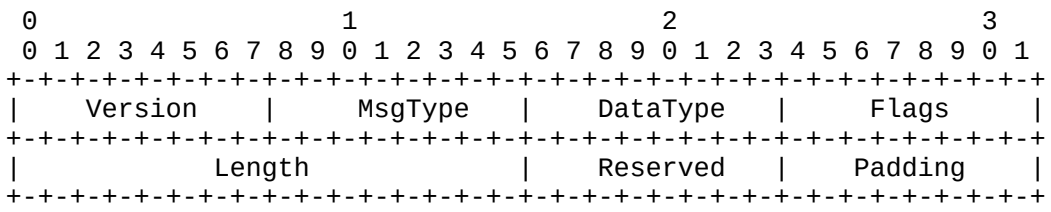

Figure 14: ASRP HS Message Format

4.1.7. PS Message Format

In PSV mode, in an asymmetric routing environment, when the server sends its first packet, it uses a PS message to push session information to the network node, enabling rapid session establishment and fast forwarding of the server’s initial packet.

PS Message Format: The structure of message is the same as that of the NS message.

4.2. ASRP packet Format

A packet that carries a single ASRP message is referred to as an ASRP packet. Based on the three transmission modes of ASRP messages, the format of ASRP packets can also be classified into three types.

4.2.1. Inline-ASRP packet

In Inline mode, the ASRP packet format is as follows:

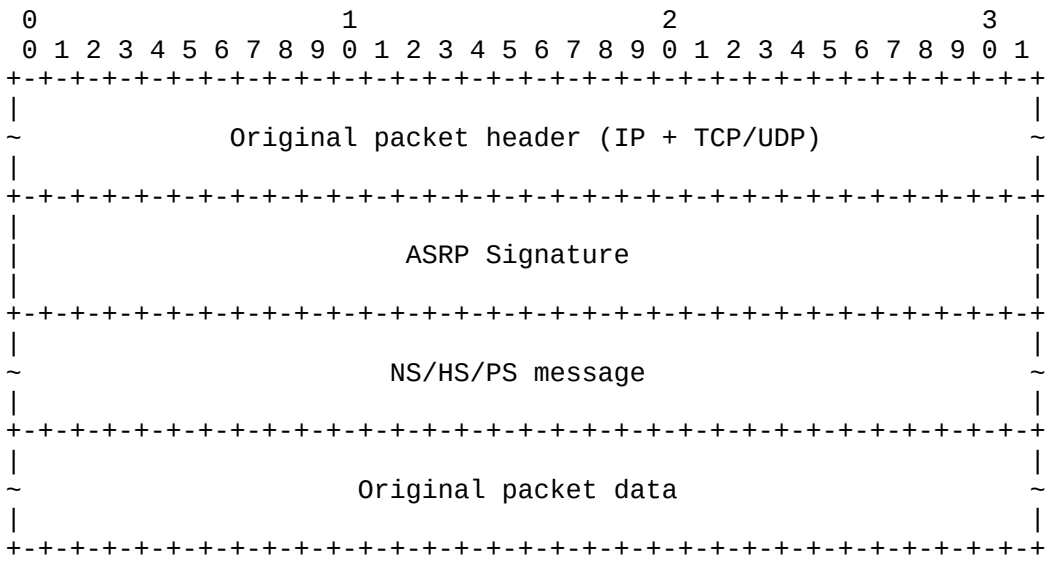

Figure 15: Inline-ASRP packet

4.2.2. Standalone-ASRP packet

In standalone mode, the ASRP packet format is as follows:

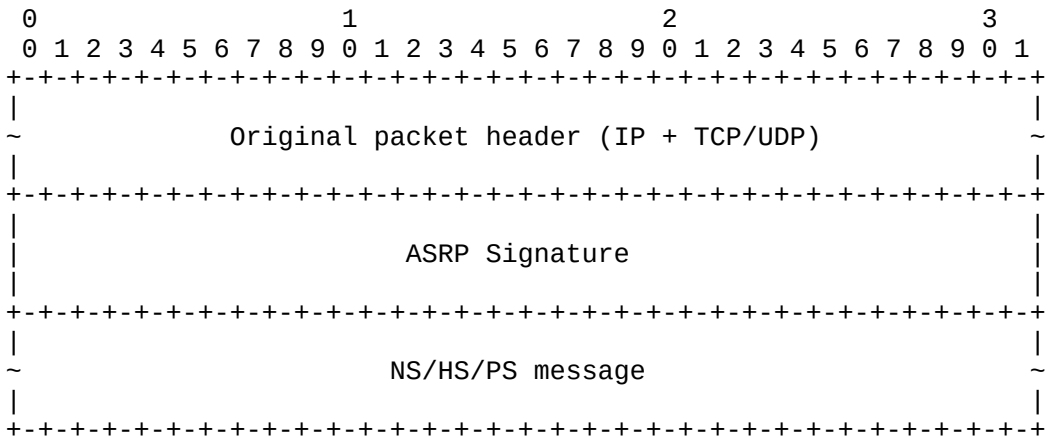

Figure 16: Standalone-ASRP packet

The message within the packet MUST have the F_MSG flag set to indicate that this is a pure message packet.

4.2.3. Bundled-ASRP packet

In bundled mode, the ASRP packet format is as follows:

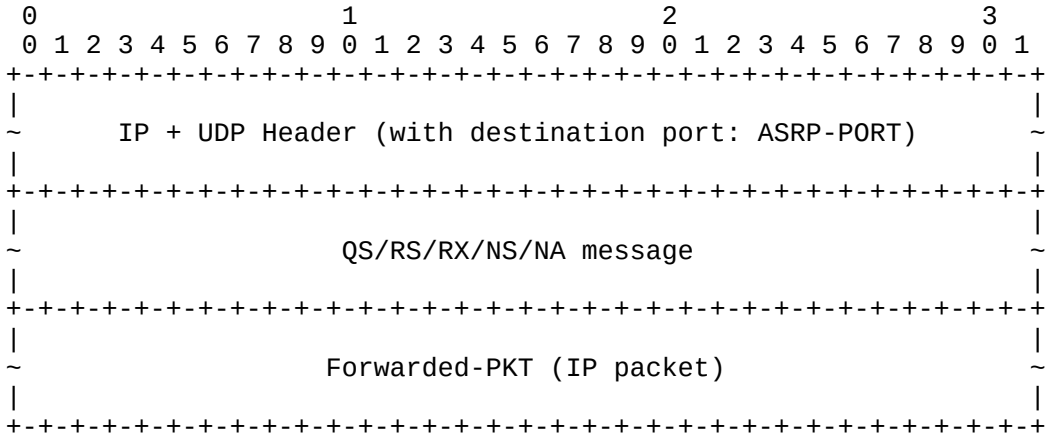

Figure 17: Bundled-ASRP packet

If there is no original IP packet following the message in the packet, the F_MSG flag MUST be set in the message to indicate that this is a pure message packet.

4.3. Message Processing

Bundled-ASRP packets are identified by the UDP destination port, while other ASRP packets are identified by the ASRP Signature. Once an ASRP packet is identified, the ASRP messages within the packet can then be parsed and processed.

If transmission can be performed without causing IP fragmentation, all ASRP messages may be transmitted together with the forwarded packet. The specific encapsulation method is defined in the ASRP Packet Format. In subsequent message processing descriptions, this point will not be repeatedly emphasized.

4.3.1. NS Message Processing

The NS message is generated by a network node when creating a new session and is used to back up the session to the client or server.

If the NS message is received via a Bundled-ASRP packet, the server MUST immediately respond with an NA message.

The source IP of the NS packet is set to the network node’s local IP (which can be obtained from configuration), and the destination IP is set to the client’s or server’s IP (which can be obtained from the forwarded packet). The source port is randomly generated, and the destination port is set to ASRP-PORT.

When a client or server receives an NS packet, it extracts the NS message and backs up the session state information, extracts the forwarded packet (if present), and hands it over to the system.

In PSV mode, if an NS message is lost, for TCP connections, the retransmission of the SYN packet will trigger the retransmission of the NS message. For other types of connections, subsequent packets will continue to generate NS messages until an NA message is received.

In ACT mode, if an NS message is lost, subsequent packets sent by the client will generate HS messages, prompting the network node to retransmit the NS message in response to these subsequent HS messages.

NS messages may be generated in both PSV and ACT modes. The handling procedures are described in Figure 5, Figure 7, and Figure 8.

4.3.2. NA Message Processing

Upon receiving an NA message, the network node MUST NOT send any further NS messages to the server.

4.3.3. QS Message Processing

The QS message is generated by the network node to query backup session state information.

The source IP of the QS packet is set to the network node’s local IP (obtainable from configuration), and the destination IP is set to the client’s or server’s IP (obtainable from the forwarded packet as described in Figure 6 and Figure 9, or derived via algorithmic mapping to the client or server as described in Figure 7 and Figure 8). The source port is randomly generated, and the destination port is set to ASRP-PORT.

When a client or server receives a QS packet, it extracts the QS message, queries the backup session state information, and returns an RS message; it extracts the forwarded packet (if present), processes it first according to the backup session state information, and then hands it over to the system.

If a QS message is lost, subsequent packets will trigger the generation of new QS packets, continuing the attempt to recover the session.

QS messages may be generated in both PSV and ACT modes. The handling procedures are described in Figure 6, Figure 7, and Figure 9.

4.3.4. RS Message Processing

The RS message is generated by the client or server in response to an QS message. It is processed by the network node to recover a session.

The RS packet reuses the protocol header of the QS packet, with the source and destination IP addresses and UDP ports swapped.

When a network node receives an RS packet, it extracts the RS message and recovers the session (upon successful QS query); it extracts the forwarded packet (if present) and forwards it according to the session.

If an RS message is lost, subsequent QS messages will continue the attempt to recover the session, thereby triggering retransmission of the RS message.

RS messages may be generated in both PSV and ACT modes. The handling procedures are described in Figure 5, Figure 6, Figure 7, Figure 8, and Figure 9.

4.3.5. RX Message Processing

The RX message is generated by either the client or the server in response to a QS message, indicating that no session was found. Upon receiving an RX message, the network node follows the processing procedures described in the respective scenarios under PSV/ACT mode.

4.3.6. HS Message Processing

The HS message is sent by the client during the initial connection establishment phase to announce to the network node that it requires an NS message to back up session state information.

The source IP, destination IP, and source port of the HS packet are copied from the packet sent by the client.

When a network node receives an HS message, it extracts the HS message, creates a session, forwards packets according to the session, and returns a pure NS packet to the client.

HS messages are only generated in ACT mode. The handling procedure is described in Figure 8.

4.3.7. PS Message Processing

The PS message is generated by the server after receiving an NS message. When the server sends its first response packet to the client, it uses the PS message to push session state information to the network node. This is used to recover the network node’s session in the case of asymmetric routing.

The source IP, destination IP, and source port of the PS packet are copied from the packet sent by the server.

When a network node receives a PS message, it extracts the PS message and recovers the session; it extracts the forwarded packet (if present) and forwards it according to the session.

PS messages are only generated in PSV mode. The handling procedure is described in Figure 5 Figure 7.

5. Security Considerations

5.1. Message Forgery Attacks

The security design of the ASRP protocol is based on its typical deployment model.

Deployment Boundaries and Access Control: ASRP recommends deploying network nodes and the clients or servers that back up sessions within the same trusted internal network domain. In this model, all ASRP protocol packets communicate within an internal address space. By implementing appropriate network segmentation (e.g., using firewall policies or security groups) and strictly checking the source addresses of packets, forged ASRP packets originating from untrusted external networks can be effectively prevented from reaching the target nodes.

Session Legitimacy Verification: When processing ASRP packets that may establish new sessions (e.g., HS or RS packets), network nodes SHOULD perform basic validation according to the specific policies of the upper-layer application or service. For instance, in a load-balancing scenario, a node SHOULD verify whether the session points to a known and healthy server. In a NAT scenario, it SHOULD verify whether the address translation complies with predefined rules. This prevents the establishment of illegal sessions at the application layer.

Internal Threat Assessment: Even if an attacker is located within the trusted network and can forge ASRP packets, the scope of impact is inherently limited. The attacker can only forge sessions where they themselves are the endpoint (e.g., masquerading as a client to request recovery of a non-existent connection). Such forged sessions are indistinguishable in form from sessions established through normal access. They do not directly jeopardize the security of other users or nodes, nor can they elevate the attacker’s privileges or grant access to unauthorized resources.

5.2. QS Flood Attacks

When a network node loses a session, it may generate a large volume of QS packets. If maliciously exploited or due to a malfunction, this could lead to a flood attack [RFC4987]. To mitigate such risks, implementers SHOULD consider the following protective measures:

Rate Limiting and Traffic Shaping: Each network node SHOULD implement monitoring and limiting of the rate at which QS packets are sent. A reasonable threshold (e.g., the number of QS packets allowed per second) SHOULD be set. When the rate exceeds this threshold, the node SHOULD adopt a packet drop policy, for example, discarding newly arriving forwarded packets that trigger queries. The parameters for rate limiting SHOULD be configurable to adapt to deployment environments of different scales.

6. IANA Considerations

This document defines an application-layer protocol (ASRP). The protocol message types and internal identifiers are defined by this specification itself and constitute internal implementation details of the protocol. Therefore, there is no need to request registration of a separate protocol number or code point from IANA. However, for the implementation of this protocol, a UDP destination port requires allocation:

6.1. UDP Destination Port

NS/QS/RS messages are encapsulated within UDP datagrams for transmission. A fixed UDP destination port number is required so that the receiving end can identify and process such encapsulated packets.

Service Name: asrp

Port Number: 51200 (proposed value for current experimentation)

Transport Protocol: udp

Description: Used for receiving UDP-encapsulated ASRP protocol messages.

For experimental implementations and interoperability testing prior to IANA assignment, UDP port 51200 MAY be used as a temporary default. This port falls within the dynamic/private port range (49152-65535) reserved for local or temporary use and documentation examples [RFC6335].

IANA is requested to assign a permanent port number in the "User Ports" range (1024-49151) for the "asrp" service in the "Service Name and Transport Protocol Port Number Registry", with a reference to this document.

7. References

7.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <https://www.rfc-editor.org/info/rfc768>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

7.2. Informative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC2991] Thaler, D. and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection", RFC 2991, DOI 10.17487/RFC2991, November 2000, <https://www.rfc-editor.org/info/rfc2991>.

[RFC2992] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", RFC 2992, DOI 10.17487/RFC2992, November 2000, <https://www.rfc-editor.org/info/rfc2992>.

[RFC4787] Audet, F., Ed. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January 2007, <https://www.rfc-editor.org/info/rfc4787>.

[RFC4987] Eddy, W., "TCP SYN Flooding Attacks and Common Mitigations", RFC 4987, DOI 10.17487/RFC4987, August 2007, <https://www.rfc-editor.org/info/rfc4987>.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, DOI 10.17487/RFC6335, August 2011, <https://www.rfc-editor.org/info/rfc6335>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/info/rfc9293>.

Appendix A. Acknowledgments

The authors would like to thank all individuals who have provided valuable feedback and contributions during the development of this document.

Authors' Addresses

Zhaoyu Luo (editor)
CMCC
No. 58 Kunlunshan Road
Suzhou
215000
China
Email: lluozy@yeah.net

Haishuang Yan
CMCC
No. 58 Kunlunshan Road
Suzhou
215000
China
Email: yanhaishuang@cmss.chinamobile.com