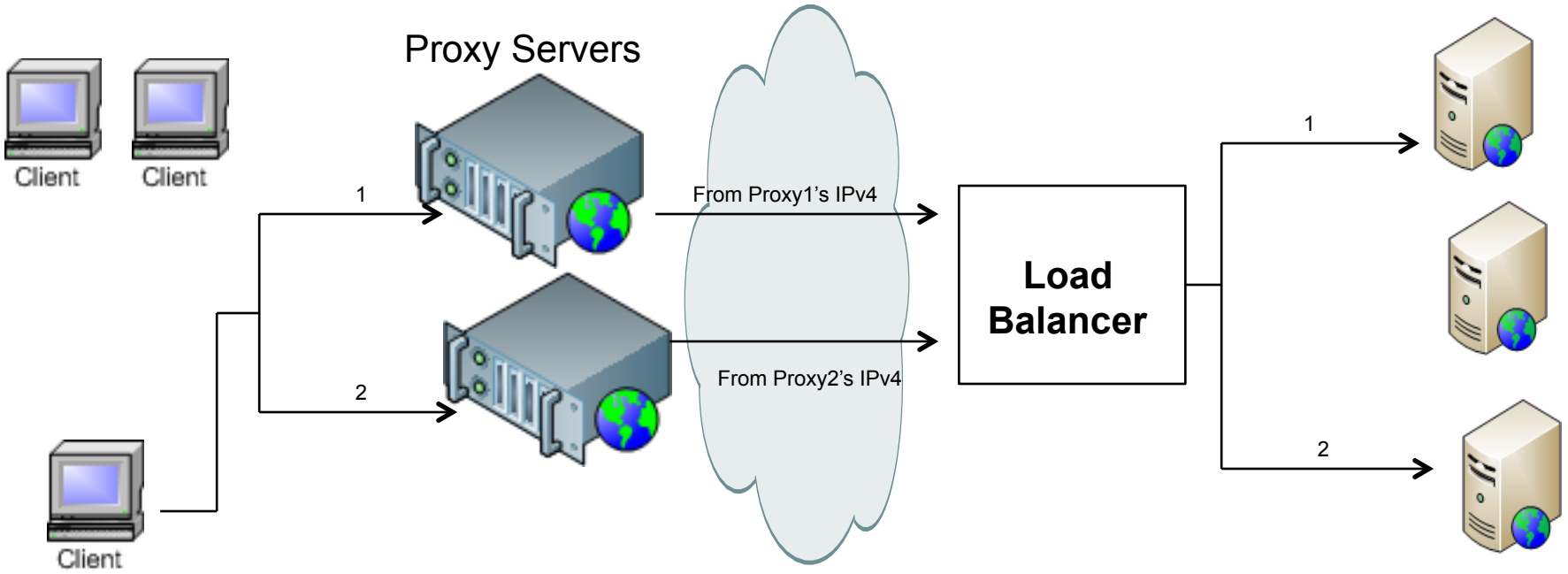
Single Pass
Load Balancing with Session Persistence
in IPv6 Network

C. J. (Charlie) Liu
Network Operations – Charter Communications

Load Balancer Today

- Load balancing is still in use today. It is now considered a feature of Application Delivery Controllers (ADC)
- In a server farm serving mainstream applications, a load balancer is employed to direct packets of an application session to a sever that is available and in healthy state
- For session persistence, all packets in the same session will go through a load balancer onto the same server
- The load balancer is always in the path of all packets from the client to the server
- The load balancer is also in the path of all packets from the server back to the client, except in the case of using DSR (direct server return) technique
- In a network topology where a load balancer is not directly connected to servers, a server farm throughput is limited by the load balancer's I/O capacity

Session Persistence Issue with MegaProxy – IPv4 Based Exchange



- When an user opens multiple connections, the connections can be distributed across multiple proxy servers
- The proxy server that makes the request to the destination web site may be different for each connection
- The LB at the destination web site see the IP address of the proxy server as the source IP address
- If the LB performs session persistence based on the source IPv4 address, the connections from the same user may be sent to different servers, causing the application transaction to break
- It can also happen that the LB directs all connections from a given proxy server to the same application server, and does not properly balance the traffic load
- Most prevalent method is for load balancer to to employ delay binding technique, and wait for a web server to set cookie in its reply using delay binding method
- Delay binding can cost LB performance in terms of the number of concurrent connections it can support, and adds latency to request-reply transaction

Megaproxy Issue in Web Applications

- Use of Megaproxy in ISPs and enterprises can lead to session persistence and load balancing problem without the use of cookie switching or URL switching in a load balancer (LB)
  - In web applications, a LB can use URL or cookie in a HTTP request to select appropriate server
- In order to examine the application packets, the load balancer (LB) must postpone the binding of a TCP connection to a server until after the application request is received – Delayed Binding
  - The LB completes the TCP connection setup before the client on behalf of the server
  - Once the HTTP request is received, the LB selects the server, establishes a connection with the server, and forwards the HTTP request
  - The LB must translate the sequence number of all reply packets from the server to match what the LB used on the client-side connection
  - The LB must also change the ACK sequence number for packets from client to the server
- Delayed Binding can impact the performance of the LB, because of the need for sequence number translation
- Delayed Binding can also increase the response time of user's request on application

E-Commerce Applications with Session Persistence

- In shopping-cart applications, the load balancer needs to find a way to associate the first HTTPS connection request to the earlier HTTP request received
  - Source IP-based persistence won't work well when dealing with mega proxy sever
  - Use cookie switching to address the mega proxy issue
  - The moment an user transitions from HTTP to HTTPS, the load balancer can no longer read the cookie because it's encrypted
- Current practices for HTTP to HTTPS transition
  - Use a shared back-end storage or database system. When the SSL session is initiated, the SSL server gets the shopping-cart information from the back-end database and processes it.
    - It requires the server with the shopping cart writes the information to a back-end database
  - Using middleware software that makes all the servers look like one big virtual server to the application.
    - A cookie is used to track the user identification. Once the app receives HTTPS request, it uses the cookie to retrieve that data structure that contains the context
  - Special configuration at the load balancer
    - Bind a different port number on the VIP to port 443 of a different real server
    - When the real server generates the Web page reply that contains the checkout button, it must link its own port number to that button (by generating hyperlink for the checkout button)
  - Using SSL Accelerator
    - A SSL acceleration product terminate SSL connections and converts the HTTPS request to HTTP request
    - The load balancer redirects requests received on port 443 to the SSL accelerator
    - Maintain session persistence via Cookie or other method that is no longer encrypted

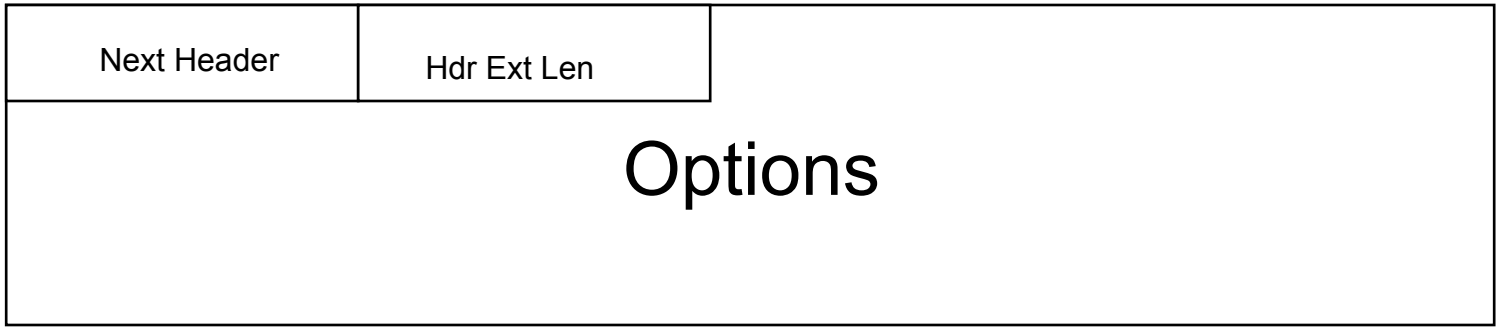
Use Destination Header for Session Persistence

- For IPv6 based application traffic, a new mechanism is available for load balancing and session persistence
- The IPv6 Destination Header can be used to minimize the need of delayed binding for session persistence, boost the performance of LB and speed up application response time
- Using IPv6 Destination Header, shopping-cart applications can carry forward the state as an user transitions from HTTP to HTTPS requests so that all connections from a given user are sent to the same server for both HTTP and HTTPS traffic
  - We can avoid the need of a back-end database, or middleware, or complicated configuration at the load balancer to achieve session persistence
  - It can help a load balancer with the same horse power to handle more connections, and reduce latency to complete transaction

New Options for IPv6 Destination Options Header

- The Destination Options header is used to carry information that needs to be examined only by a packet's destination node(s). The Destination Options header is identified by a Next Header value of 60 in the immediately preceding header. (Ref: RFC 2460)
- "LBSP Option" to be placed in Destination Options Header for every IPv6 packet exchange between a client and an application server that requires load balancing and/or session persistence for scalability, high availability, manageability, and security
  - The LBSP option data is preserved in traversing ISP or enterprise proxy servers
  - The LBSP option data is not encrypted in HTTPS request and is visible to a load balancer in HTTP to HTTPS request transition

IPv6 Destination Options Header – Format (from RFC 2460)

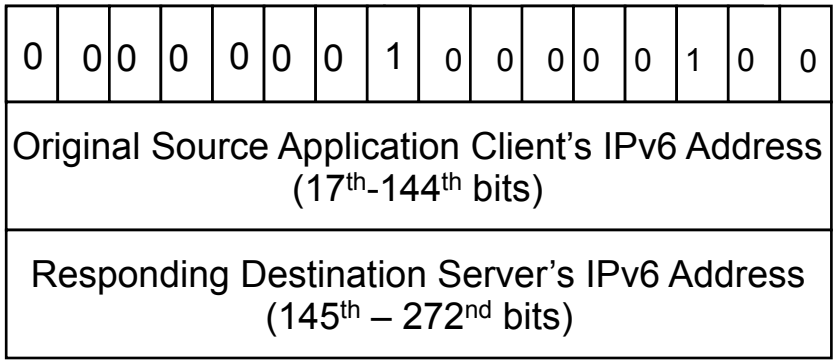


- Next Header: 8-bit selector. Identifies the type of header immediately following the Destination Options header. Uses the same values as the IPv4 Protocol field.
- Hdr Ext Len: 8-bit unsigned integer. Length of the Destination Options header in 8-octet units, not including the first 8 octets.
- Option: Variable-length field, of length such that the complete Destination Options header is an integer multiple of 8 octets long. Contains one or more TLV-encoded options.

LBSP Option

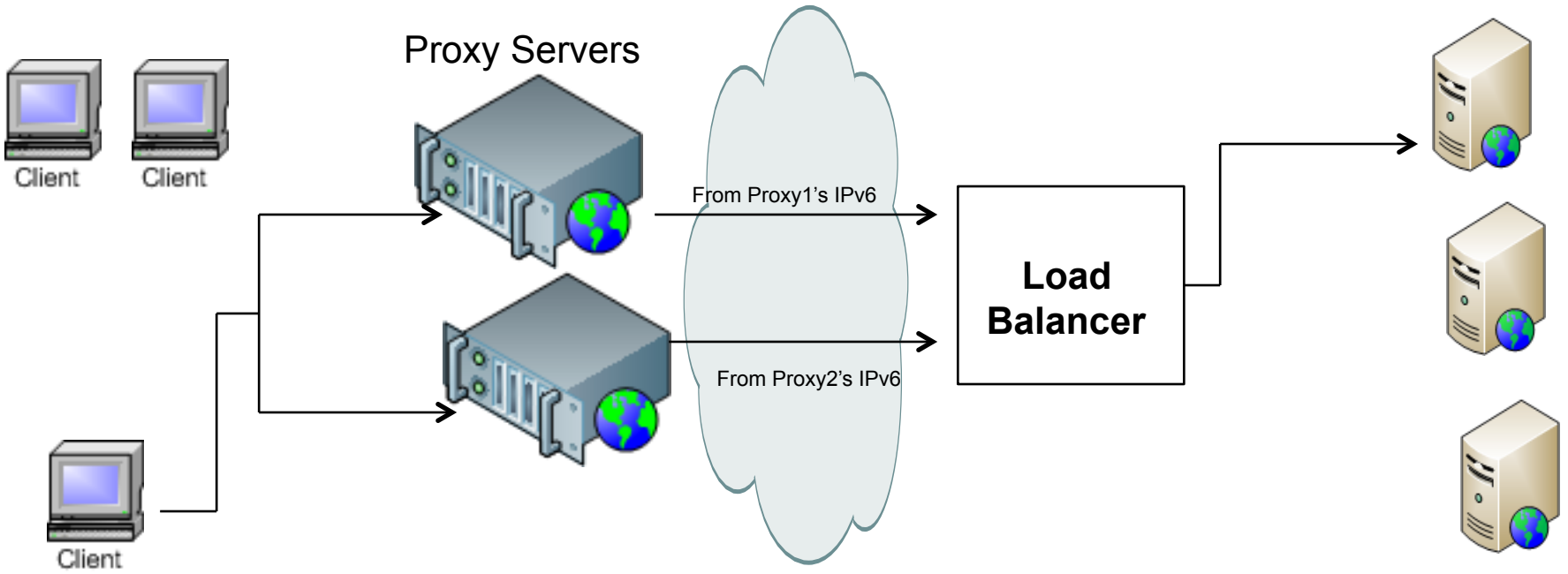
- Option Type: 1 (00000001)
- Opt Data Len: 32 bytes (00000100 in 8 octets unit)
- Option Data: IPv6 Address of the Source and the Server
  - There are two segments in the Option Data:
    - The first 16 bytes is for the Source IPv6 address
      - Alternatively, it can be a random number generated by the source application for anonymous purpose
    - The second 16 bytes is for the Server IPv6 address
  - When a source application makes a connection request for the 1st time, it'll insert its IPv6 address into the 1st segment and leave the 2nd segment all zeroes in the session persistence option of its IPv6 destination header
    - The VIP of a load balancer is the destination IPv6 address of the application request
    - The Load Balancer can then select the destination server based on the data in the 1st segment of the option data and other load balancing criteria (such as least connections, server health check, and etc.) without waiting for server to set cookie for load balancing purpose
    - Subsequent requests should include the session persistence option data with its own IPv6 address and responding server IPv6 address
  - When a server application responds to a request for the 1st time, it'll rewrite the all zeroes 2nd segment with its own IPv6 address. The server should not change the 1st segment of the option data
    - All subsequent responses to the client should also include the same session persistence option
  - A proxy server will not and should not change the session persistence option when it terminates a connection and makes the request on the user's behalf
    - The destination header should be examined only by the destination (the VIP of a load balancer), not by any intermediate node
    - Proxy server will change the source IPv6 address (9th-24th bytes in the v6 packet header) to its own, but not the content in the destination header

LBSP Option



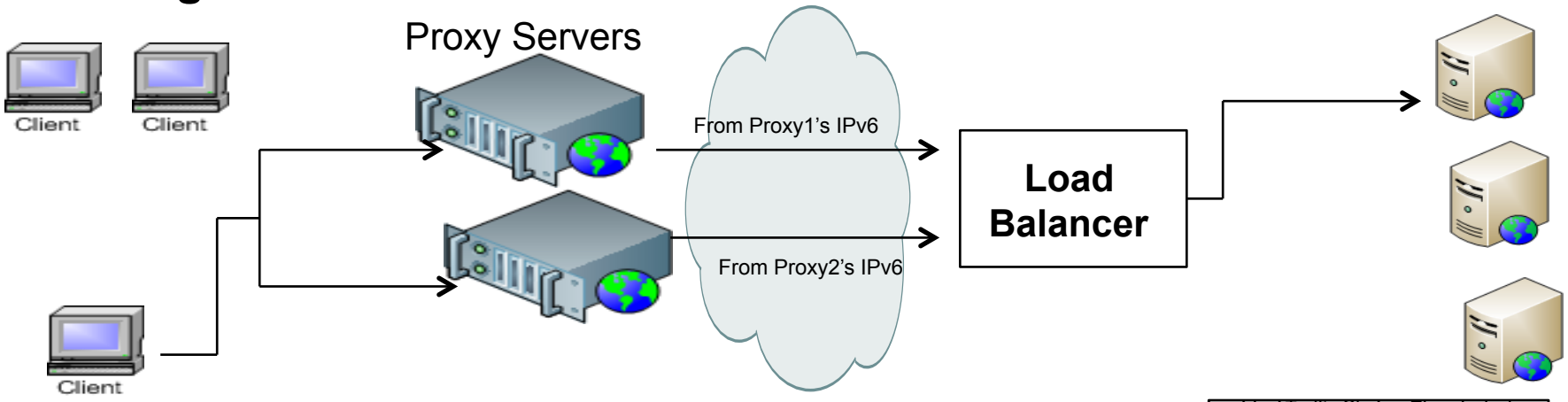
- Option Type: The 1st eight bits (value 1) identifies the session persistence option
- Option Data Length: The 2nd eight bits (value 4) indicates the length of the option data is 32 bytes
- If the session persistence option is employed by an user application, the 17th bit to the 144th bits will be the IPv6 address of the source application client (same as 9th – 24th bytes of the original IPv6 packet header)
  - A load balancer (LB) will use this Original Source Application Client IPv6 address and other criteria to pick a server for load balancing purpose
- The server chosen by the LB shall also respond with a LBSP option in its IPv6 response packet and put its IPv6 address in the segment for the Responding Destination Server IPv6 address
- Subsequent request and response packets should also include the LBSP option with both the source application and destination server IPv6 address
- The LB will use the Responding Destination Server IPv6 address for session persistence purpose
- If either AH or ESP Header is present, the Destination/LBSP option is before those headers

Session Persistence with Megaproxy – IPv6 Based Exchange -1



- If the transaction is based on IPv6 exchanges, the client's IPv6 address will be persistent and made available in the LBSP option data (LBSPOD) of each IPv6 packet header
- The LB can use the application client's IPv6 address in LBSPOD along with other load balancing criteria for initial server selection at the 1st TCP SYNC packet
- The LB can use responding server's IPv6 address and client's IPv6 address for the purpose of session persistence
- The proxy servers will and should transparently pass the Destination Option of application client requests and associated response from load balancers or directly from servers (in DSR scenarios)
- With the LB using the session persistent option data in IPv6 packet header, the need for delay binding can be minimized and therefore avoid the costly sequence number translation

Session Persistence with Megaproxy – IPv6 Based - 1st Exchange Packet

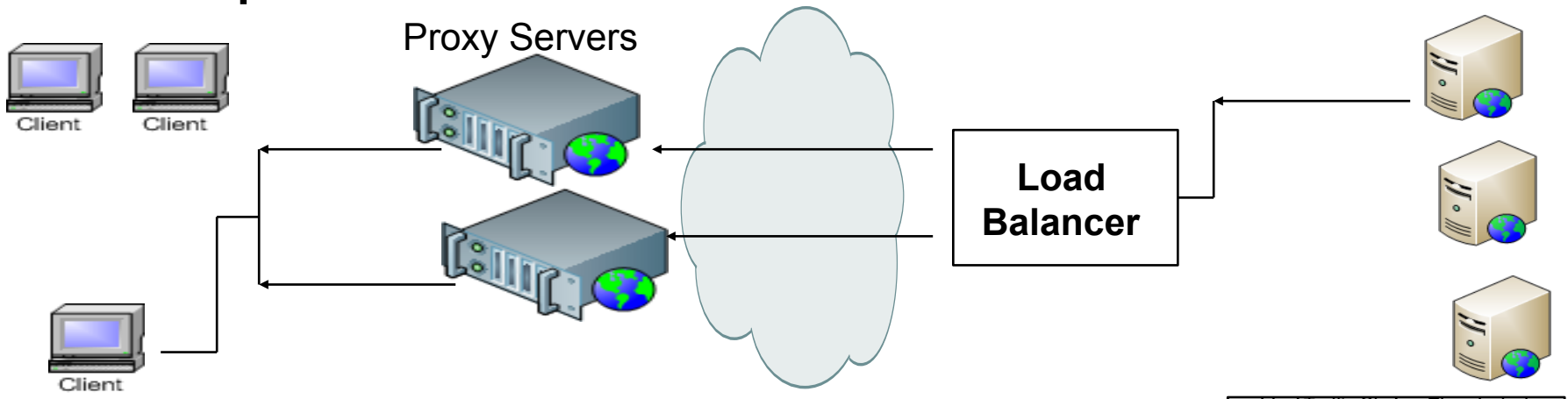


V	Traffic CL	Flow Label	
Payload Length		60	HL
Source App. Client IPv6 Address			
Destination VIP IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
All zeros for TCP Sync packet or 1st UDP packet			
Client Request Packet Data			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Proxy Server IPv6 Address			
Destination VIP IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
All zeros for TCP Sync packet or 1st UDP packet			
Client Request Packet Data			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Load Balancer VIP IPv6 Address			
Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
All zeros for TCP Sync packet or 1st UDP packet			
Client Request Packet Data			

Session Persistence with Megaproxy – IPv6 Based – Respond to 1st Request

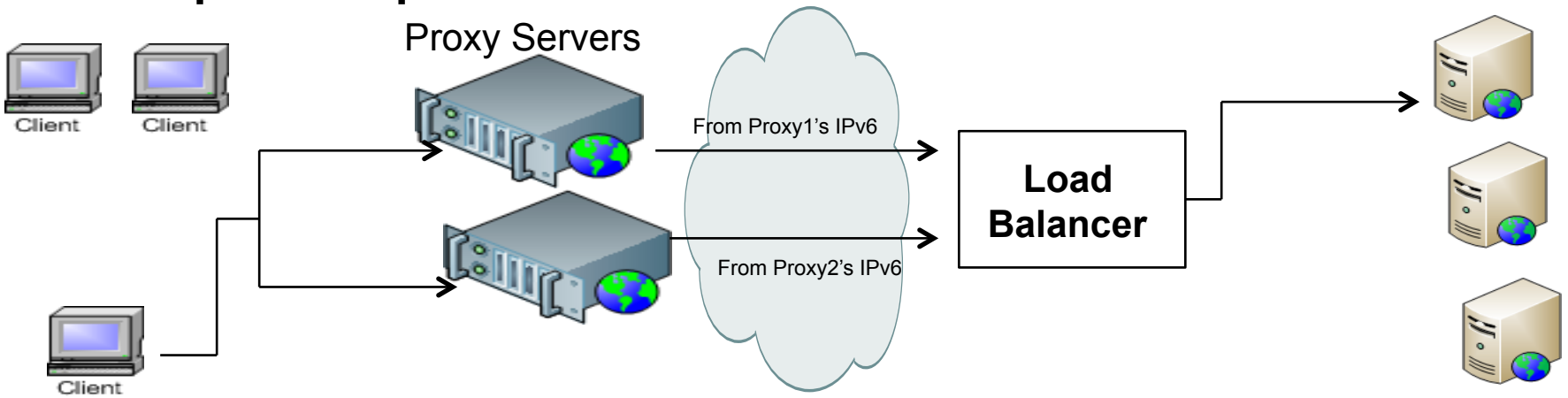


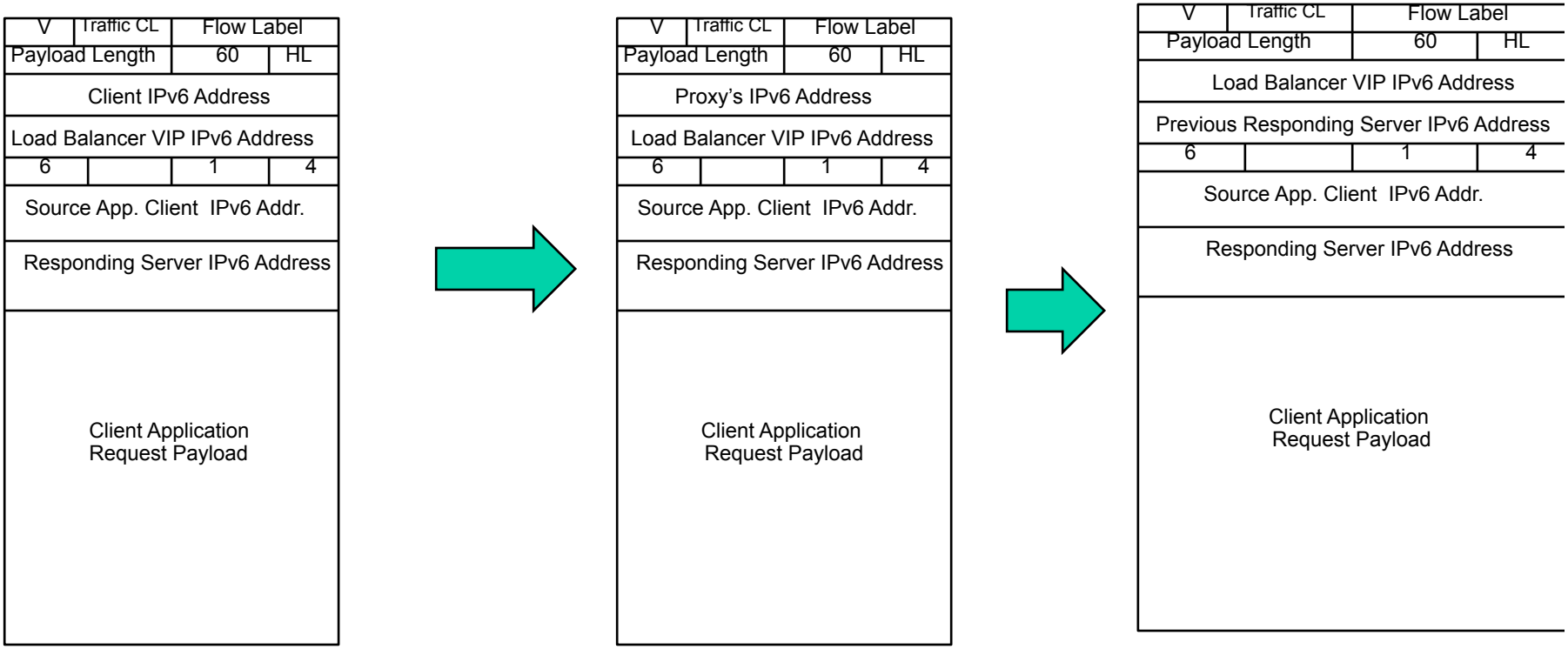
V	Traffic CL	Flow Label	
Payload Length		60	HL
Load Balancer VIP IPv6 Address			
Client IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Load Balancer VIP IPv6 Address			
Proxy Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

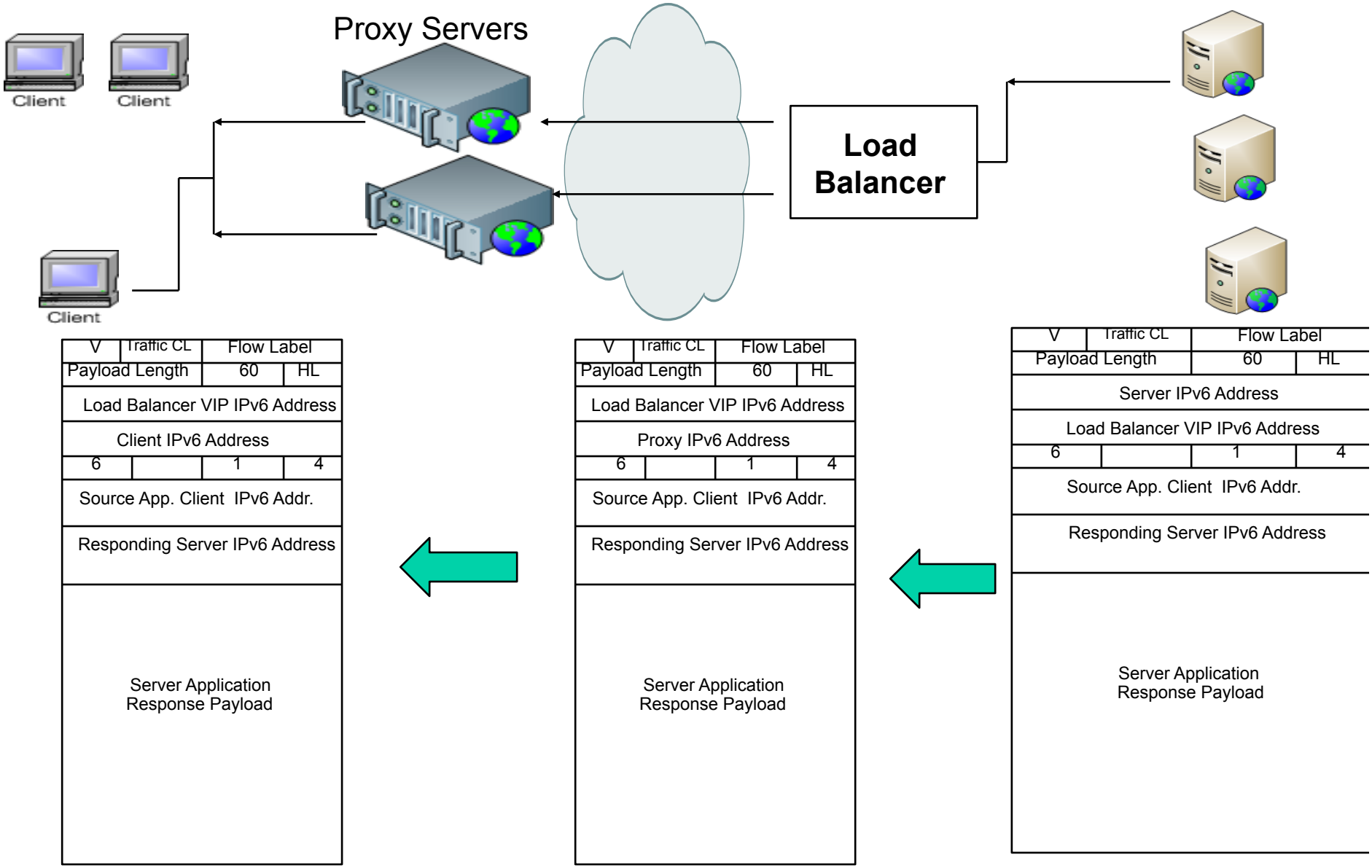
V	Traffic CL	Flow Label	
Payload Length		60	HL
Responding Server IPv6 Address			
Load Balancer VIP IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

Session Persistence with Megaproxy – IPv6 Based – Subsequent Request Packets

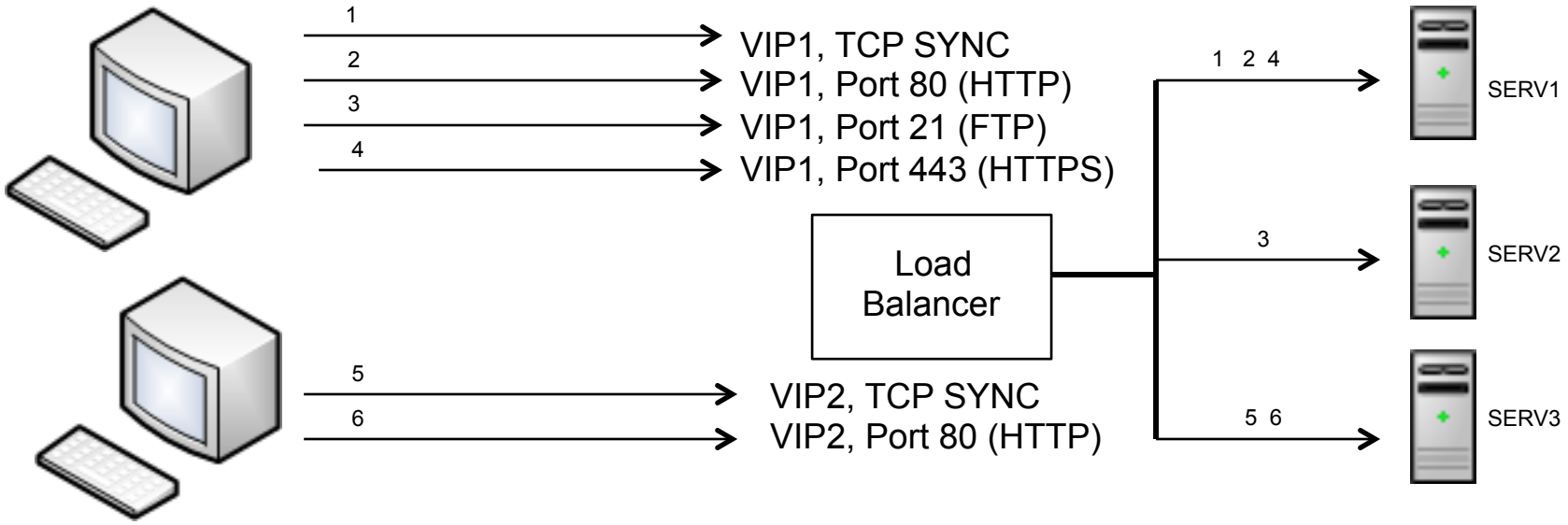




Session Persistence with Megaproxy – IPv6 Based – Subsequent Response Packets



Session Persistence at a Load Balancer



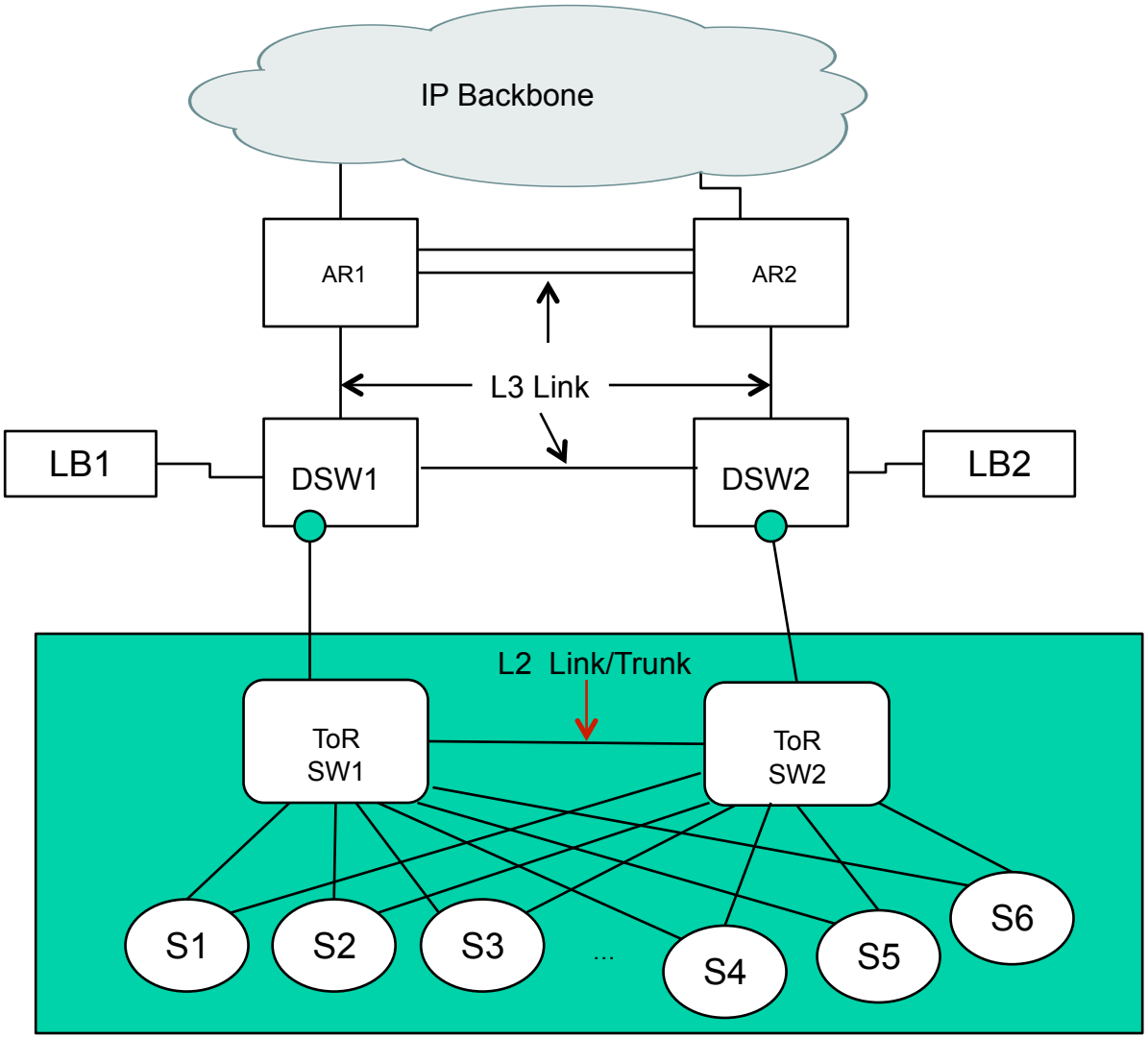
	Source	Dest	DEST Port	LBSPOD-1	LBSPOD-2	Server Selection Based on	Real Server Assigned
1	Proxy1	VIP1	80	C1	0	LB	SERV1
2	Proxy1	VIP1	80	C1	SERV1	LBSPOD-2	SERV1
3	Proxy1	VIP1	21	C1	0	LB	SERV2
4	Proxy2	VIP1	443	C1	SERV1	LBSPOD-2	SERV1
	Proxy2	VIP1	443	C1	SERV1	LBSPOD-2	SERV1
5	Proxy1	VIP2	80	C2	0	LB	SERV3
6	Proxy1	VIP2	80	C2	SERV3	LBSPOD-2	SERV3

- LBSPOD-1 is the segment for the client's IPv6 address in the Session Persistence Option of the destination header
- LBSPOD-2 is the segment for the responding server's IPv6 address in the Session Persistence Option of the destination header
- Above table assumes the traffic went through an ISP or enterprise proxy server before reaching the load balancer site

One Step Further – Single Pass LB

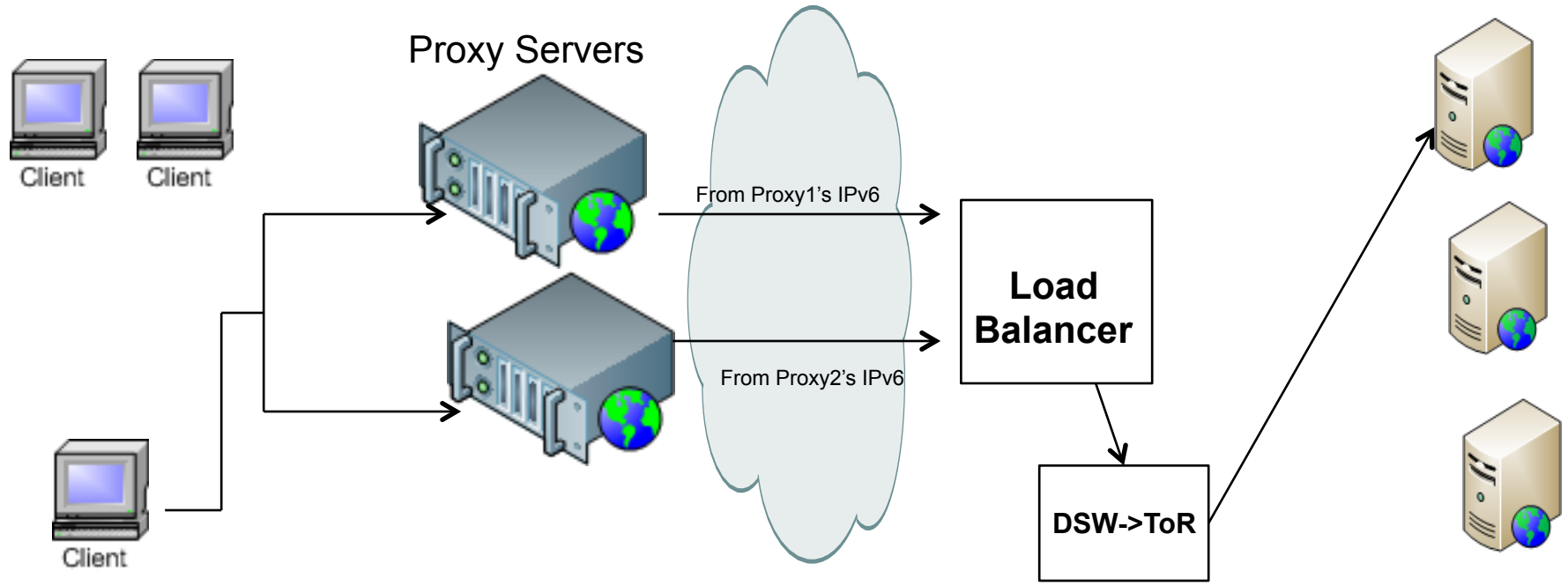
- For IPv6 based application traffic, a new mechanism is proposed to allow the load balancer out of the traffic flow after the application session is established
- Using IPv6 Destination Header to eliminate the need of delayed binding for session persistence, boost the performance of LB and speed up application response time
- For ISP planning to use public routable IPv6 address for servers
- It's proposed that client (such as a web browser) takes advantage of the information of the IPv6 address of the serving server in the IPv6 destination header of returned packets, and send subsequent packets with the serving server's IPv6 address as the destination address
  - i.e. the client will no longer use the load balancer's VIP as packets' destination address after the application session is established
- Benefits
  - A load balancer with the same horse power and I/O bandwidth will be able to handle more connections,
  - The throughput of a server farm will increase

Data Center Topology - LB Not in Direct Path



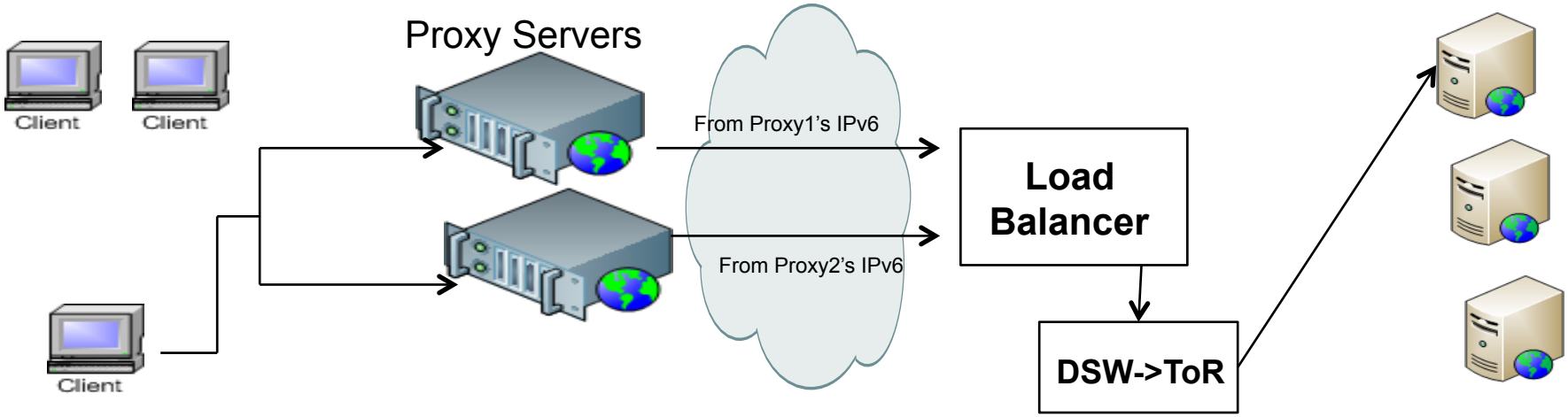
- LB : Load Balancer
- ToR : Top of Rack

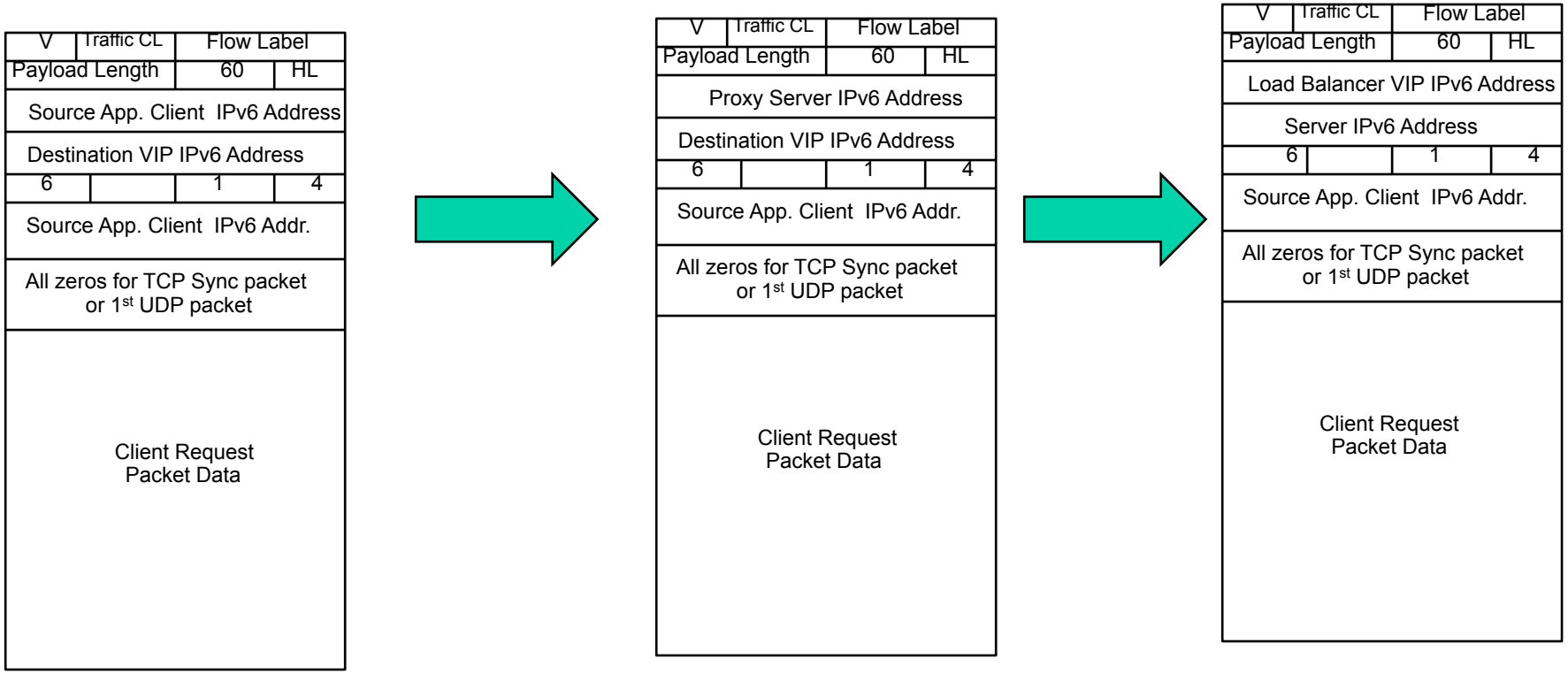
IPv6 Based Exchange Overview



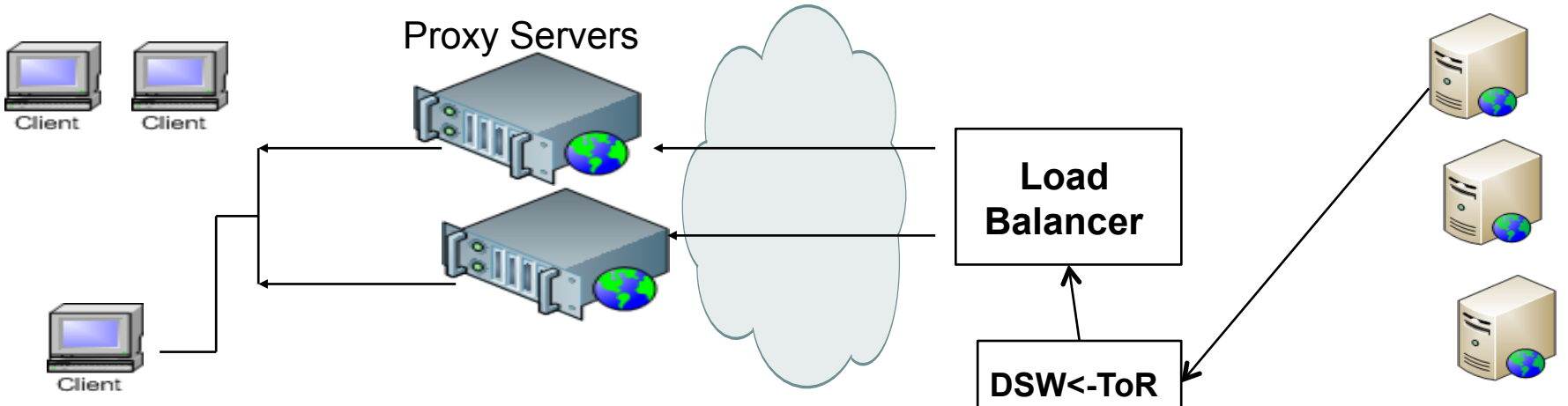
- Servers will use public IPv6 addresses
- 1st packet will be sent to IPv6 VIP of load balancer for server selection
- The LB can use the application client's IPv6 address in LBSPOD along with other load balancing criteria for initial server selection at the 1st TCP SYNC packet
- Server will inform client its own IPv6 address via LBSPOD in IPv6 destination header
- The client will send packets directly to server after learning responding server's public IPv6 address
- The proxy servers will and should transparently pass the Destination Option of application client requests and associated response from load balancers or directly from servers (in DSR scenarios)
- With the LB using the session persistent option data in IPv6 packet header, the LB does not have to use delay binding and therefore avoid the costly sequence number translation

Session Persistence– IPv6 Based - 1st Exchange Packet





Session Persistence– IPv6 Based – Respond to 1st Request

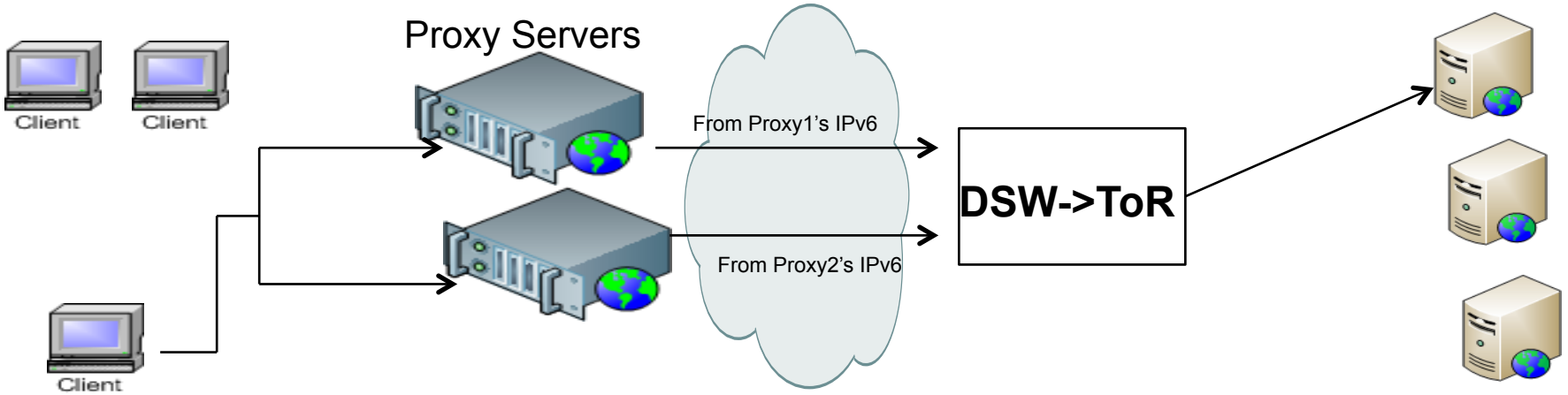


V	Traffic CL	Flow Label	
Payload Length		60	HL
Load Balancer VIP IPv6 Address			
Client IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Load Balancer VIP IPv6 Address			
Proxy Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Responding Server IPv6 Address			
Load Balancer VIP IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Response Packet Data			

Session Persistence– IPv6 Based – Subsequent Request Packets

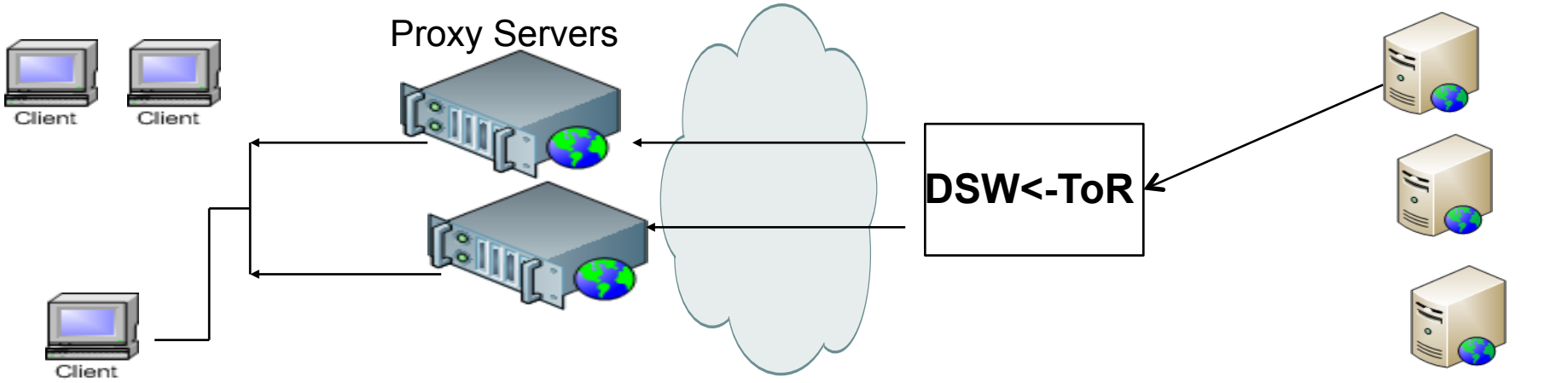


V	Traffic CL	Flow Label	
Payload Length		60	HL
Client IPv6 Address			
Serving Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Client Application Request Payload			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Proxy's IPv6 Address			
Serving Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Client Application Request Payload			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Proxy's IPv6 Address			
Serving Server IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Client Application Request Payload			

Session Persistence– IPv6 Based – Subsequent Response Packets



V	Traffic CL	Flow Label	
Payload Length		60	HL
Serving Server IPv6 Address			
Client IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Application Response Payload			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Serving Server IPv6 Address			
Proxy IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Application Response Payload			

V	Traffic CL	Flow Label	
Payload Length		60	HL
Serving Server IPv6 Address			
Proxy IPv6 Address			
6		1	4
Source App. Client IPv6 Addr.			
Responding Server IPv6 Address			
Server Application Response Payload			

Requirements for Application Client at the Source -1

- Use IPv6 Destination Header
- In the first packet of web or other applications, insert its IPv6 address in the 1st segment of the Session Persistence Option Data of the Destination Header, and leave the 2nd segment all zeroes
- Read the Responding Server IPv6 address in the response packet, and cache the information for subsequent IPv6 packets exchange for the same session
- For subsequent exchange, the IPv6 packets sent from the client will use the responding server's IPv6 address as the destination IPv6 address
  - The IPv6 packets shall include the Destination Header with both the client's and responding server's IPv6 address in the Session Persistent Option Data
- If the returned packet does not include the responding server's IPv6 address in the LBSPOD or destination header altogether, the application client shall use the VIP of the load balancer as a result of DNS query
  - The process is illustrated in next slide

Application Client at the Source -2