Array NETWORKS

White Paper

# Connection Multiplexing

## Understanding Array TCP Offload Technology

APV Series Application Delivery Controllers

## Overview

When the number of connections coming to a server increases, eventually the server runs out of network resources. This does not necessarily mean the server doesn't have enough horsepower, memory, or bandwidth available, it simply means that the server's TCP/IP implementation wasn't meant for such a high rate of connection turnover. Enter connection multiplexing. Learn about:

- How connection multiplexing converts a large number of short connections to a lesser number of high-throughput connections
- Using connection multiplexing to take advantage of optimized bulk throughput server settings without changing configurations or content
- Doing more with less by implementing connection multiplexing technology

## Introduction

It was a Friday afternoon that the problem first came up. Array Networks had just moved into their first real office, a mere two months after being born, when an engineer wanted to show off his latest performance improvements. Unfortunately, he found himself frustrated that the few servers he had were not enough to demonstrate the increases.

The engineer spoke with Array Networks' founder and CTO, about purchasing additional servers. With tight budgets and no outside investors yet, he had to say no to the request for additional servers.

Thinking about the problem some more, he realized that his future customers would face similar situations as well. Going on an all-weekend hacking spree, the two designed and implemented a new feature: Connection Multiplexing.

With the new feature in place, the CTO and his engineering team were able to demonstrate significant performance improvements using the same handful of servers built out of leftover parts.

Marketing was impressed. A new feature was born.

## How Connection Multiplexing Works

Typical network topologies put Array APV Series application delivery controllers (ADCs) between Web clients and Web servers. In these situations, the APV appliance acts as an intermediary by accepting connections on behalf of the servers and then passing the connection onward if it meets the necessary security criteria. (i.e. the packet is not part of a DoS attack, etc.).

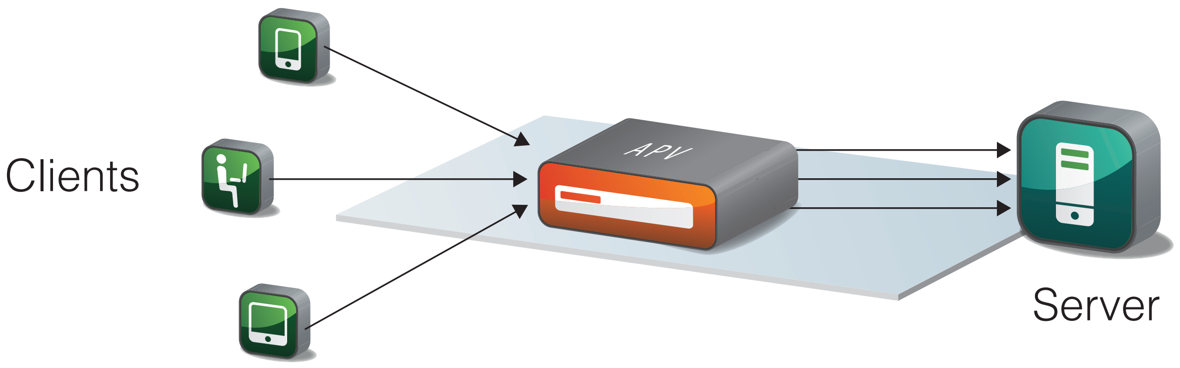

*Figure 1: Array ADC configuration showing one-to-one connections*

## Understanding Array TCP Offload Technology

The result of such a configuration is a one-to-one correlation between connections sent by users and connections received by Web servers. While this may work okay in low-volume traffic, servers are often not designed to support a large number of concurrent connections with a high rate of connection turnover.

When the number of connections coming to a server increases, the server eventually runs out of network resources. This does not necessarily mean the server doesn't have enough horsepower, memory, or bandwidth available, it simply means that the server's TCP/IP implementation wasn't meant for such a high rate of connection turnover.

This situation is made worse by the fact that most TCP/IP stacks are not tuned for a large number of short connections, but rather for bulk throughput.

Enter connection multiplexing.

The purpose of connection multiplexing is to convert a large number of short connections into a few connections needing more throughput. This allows us to take advantage of the server's optimized bulk throughput settings without changing any configurations or content.

Connection multiplexing works by taking advantage of a feature in HTTP/1.1 that allows for multiple HTTP requests to be made over the same TCP connection. So instead of passing each HTTP connection from the client to the server in a one-to-one manner, the APV application delivery controller combines many separate HTTP requests from clients into relatively few HTTP connections to the server. The APV keeps the connections to the server open across multiple requests, thus eliminating the high turnover typically encountered in high-volume web sites.

Net result: higher performance out of the same servers *without* any changes or improvements to the server infrastructure.

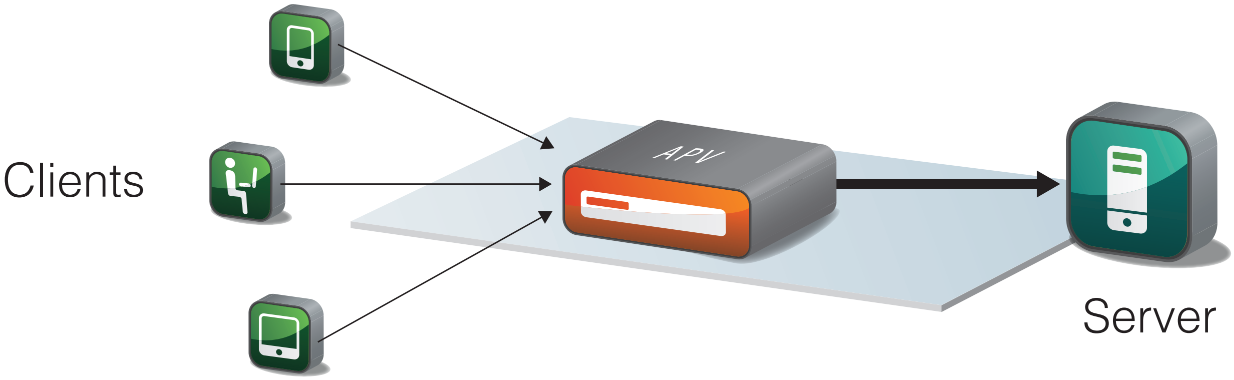

***Figure 2: Array ADC configuration showing many-to-one connection***

## Technical Details of Connection Multiplexing

Array application delivery controllers work as a full proxies, thereby allowing them to perform connection multiplexing. With this feature, the Array ADC pre-opens several connections to the origin server and keeps them open. As connections come from various clients, the ADC attempts to reuse the pre-opened connections.

The Array ADC will keep the connection to the server open until it receives a "`Connection: close`" HTTP header or TCP FIN packet from the server.

Because of connection multiplexing, the Array ADC is able to take advantage of the Van Jacobson Header Prediction found in most TCP stacks (including Microsoft Windows 2000/.Net, Solaris, Linux, and the many flavors of BSD Unix). This feature of TCP stacks provides a fast path to move packets from the core operating system (kernel space) to the Web server application itself (user space) so long as each packet meets the following requirements:

- The packet's associated connection is established
- The SYN, FIN, RST, or URG flags must not be set
- The packet must acknowledge previously sent data
- The packet is in order
- The TCP window size has not changed

Tests performed by W. Richard Stevens (author of the TCP/IP Illustrated Series, the now *de facto* definition of TCP/IP) found that header prediction worked between 97 and 100% of the time in LAN environments, which is typically the case between Array ADCs and servers.

Another important benefit of connection multiplexing is that it removes the burden placed on servers to turn over new connections very frequently – a process whose code path is often not as optimized as the path of a packet that meets the header prediction requirements. Furthermore, by not having to turn over new connections, the server is less likely to burn up its ephemeral ports by placing them into the TIME_WAIT state.

Because connection multiplexing removes the one-to-one correlation between client connections and server connections, the application delivery controller inserts the X-Forwarded-For HTTP header into each request. This header has one parameter, the client IP's address. Modules for IIS and Apache Web servers are available from Array Networks to set the client IP's address in the necessary data structures, logging systems, and CGI environment variables. These modules are available with source code to any Array Networks customer.

If the administrator chooses not to have connection multiplexing enabled, the APV can transparently pass the client IP address in the IP header of each packet, thereby allowing the origin server to work as if it were directly communicating with the client browser.

Note that connection multiplexing and client IP transparency are mutually exclusive. This is because it is not possible to spoof the client IP address from one client, keep the same connection open, and then spoof another client IP for a second request. The best alternative is the insertion of the X-Forwarded-For header with the true client IP on a per-request basis.

## Summary

Connection multiplexing is an easy way to reduce load on Web servers without needing to make any drastic infrastructure changes to either the network or the servers. Array Networks makes this technology easy to deploy and easy to administer in all application delivery controller products.

## About Array Networks

Array Networks is a global leader in application delivery networking with over 5000 worldwide customer deployments. Powered by award-winning SpeedCore® software, Array application delivery, WAN optimization and secure access solutions are recognized by leading enterprise, service provider and public sector organizations for unmatched performance and total value of ownership. Array is headquartered in Silicon Valley, is backed by over 250 employees worldwide and is a profitable company with strong investors, management and revenue growth. Poised to capitalize on explosive growth in the areas of mobile and cloud computing, analysts and thought leaders including Deloitte, IDC and Frost & Sullivan have recognized Array Networks for its technical innovation, operational excellence and market opportunity.

**Corporate Headquarters**
info@arraynetworks.com
408-240-8700
1 866 MY-ARRAY
www.arraynetworks.com

**EMEA**
rschmit@arraynetworks.com
+32 2 6336382

**China**
support@
arraynetworks.com.cn
+010-84446688

**France and North Africa**
infosfrance@
arraynetworks.com
+33 6 07 511 868

**India**
isales@arraynetworks.com
+91-080-41329296

**Japan**
sales-japan@
arraynetworks.com
+81-44-589-8315

To purchase
Array Networks
Solutions,
please contact your
Array Networks
representative at
1-866 MY-ARRAY
(692-7729) or
authorized reseller.

Feb-2017 rev. a