# The Applications Doctor

## WebMux Network Traffic Manager

## Introduction White Paper


**Manage, Control, and Secure Local Network Traffic for**
**High Availability of Applications and Services**

**May, 2017**

**www.avanu.com**

# Notices and Contact Information

## Copyrights



## Trademarks and Service Marks

AVANU, AVANUAdvantage, Flood Control, MAP, WebMux are trademarks or registered trademarks of AVANU, Inc.

'It is all about the user experience on your network and keeping everyone connected' is a service mark of AVANU, Inc.

This document identifies product names and services known to be trademarks, registered trademarks, or service marks of their respective holders. They are used throughout this document in an editorial fashion only. Use of a term in this document should not be regarded as affecting the validity of any trademark, registered trademark, or service mark. AVANU, Inc. is not associated with any product or vendor mentioned in this document.

## Update Information

All products and specifications are subject to change without notice. Contact AVANU, Inc. for the latest information regarding all product and services.

## Contact Information

AVANU, Inc.
1.888.248.4900 U.S. Toll Free
1.408.248.8960 International
Email: info@avanu.com
Web Site: http://www.avanu.com

# Table of Contents

# White Paper Introduction

## Troubled Waters

The revolutionizing Internet on hand has explored new horizons with leaps and bounds, while on the other hand the Internet is packed with performance related problems. Let's review the major shortcomings and how to overcome them so that you are empowered with the turbo packed performance.

The Internet is compiled of different fabrics and connection points all of which exhibit unpredictable traffic patterns, latency, and varying packet loss.

Applications that were built for high-speed Internet links are now used over bandwidth constrained links resulting in high latency and packet loss.

As the fabric of Internet was built without performance in mind, the packets will eventually get to their destination and you will ultimately be able to view a web page. This style of connectivity forms the culture to our communication, yet it essentially just works.

Over the last decade, we have seen substantial changes in the application and there is an increasing demand to support real-time interaction and multimedia content. Traffic volumes are mushrooming and the application requirements are getting more sophisticated and stringent.

Now, the question that surfaces is how does the Internet cope with the new era of application delivery, consumption, and availability? The primary aim is to send data as efficiently, infrequently, and as little as possible.

To efficiently host a web application requires more than just fast hardware. It requires a planned and a well organized model for reliability, scale, and performance. To be precise, we need an appliance that bridges the gap between the poorly performance-oriented protocols and the application that sits on top. This device would intercept the requests dynamically and would enhance the efficiency of the application response times by using a variety of techniques.

This role is fulfilled by a load balancer and over the course of the last decade it has naturally transitioned into what's commonly known as an Application Delivery Controller (ADC).

As the name suggests, the load balancing allows you to boost the performance of your entire infrastructure and not just one machine. They manage application traffic in many ways to avoid the performance degradation of a single server failure.

Traditionally, in the application world, it's common and a requisite to scale up. Scale up is performed by adjoining additional hardware components such as Random Access Memory (RAM) and Central Processing Unit (CPU); however, eventually you will hit hard limits. Scaling out requires a different approach and allows you to fulfill a bona fide type of requirement that automates redundancy.

These enhancing designs may spring certain challenges such as how to manage and operate a single **cohesive application** over a group of individual machines. However, this is the prime objective of load balancing; it makes many resources appear as one and manages the traffic in an intelligent way.

Load balancers are a standard way of doing things these days and are an essential component in today's application world. The 'modern day' computing is said to be unfinished without employing an appropriate load balance to do the task. They are invaluable assets for any organization hosting Web applications. The world without load balancers would be a pretty sluggish one.

No organization would like to invest on an elephant when a horse is in demand. In addition to the performance, cost-effectiveness is another mandatory parameter listed in the purchase rule book of a buyer. In other words, the load balancer should not only balance the load of your Internet efficiently but it should also not cause a dent in your wallet.

So, the next question that spawns is; which load balancer would be ideal for you? Going forward, you will be able to unearth an answer to all such questions like which, why, and how. But before we proceed, let's take a sneak-peek at the AVANU background and fidelity.

## Introducing AVANU

AVANU is a leading provider of both physical and software IP-based WebMux load balancers. Their product set is a flawless amalgamation of low cost and high-performance. The WebMux load balancers are easy-to-deploy with solid-state-assured high reliability thereby ensuring optimum performance.

The assembly is performed in the USA, incorporating top-quality components manufactured by authenticated US companies. AVANU strives for high reliability, feature rich, and throughput at an affordable price. It is a complete solution with no additional licensing required to unlock the advanced features.

The WebMux code is in development since 1987 and has been certified by globally renowned companies like Microsoft® and Oracle®. The load balancing platforms deliver unmatched reliability with the industry's lowest total cost of ownership.

The product set supports customer regulatory compliance requirement, including FIPS 140-2 validated encryption, security patches, Trade Agreements Act (TAA), and Payment Card Industry (PCI) compliance.

AVANU has a dedicated Research and Development team. The team is continually working to enhance the firmware to support tailored customer needs for additional feature enhancements and compliance requirements.

The product range for load balancers fit every type of organizational environment. All product variations are equipped with the same feature functionality set but vary in max throughput and processing capabilities. This makes them a leading supplier of load balancers for both small and large organizations.

### Benefits of AVANU Load Balancers

The WebMux is a server load balancing and traffic management network appliance. It incorporates key networking functionality into a rack-mountable 1U form factor appliance, including Layer 4 through Layer 7 load balancing and advanced traffic management, Secure Sockets Layer (SSL) accelerating, distributed denial-of-service (DDoS) protection, and other security services. The WebMux load balancer manages incoming client traffic and directs to a pool so that no one server is overloaded. An optimized load balancing algorithm uses minimal overhead, requiring no software interfacing or other resource contention.

The granular automatic health checks automatically evaluate the functionality of the servers in the pool. If a problem is identified, the traffic client traffic is automatically directed to another available server.

Real servers are added and drained from the server farm without disrupting existing sessions. This type of functionality enables non-intrusive maintenance windows while increasing network capacity and application scalability at the same time.

Improved scalability allows the ability to scale back-end resources with ease. Trying to scale applications with individual code configuration is a challenge. It's far more commendable to allocate a load balancing device that sits outside the application logic to perform this type of functionality.

The solid features and high performance of the AVANU product sets help organizations overcome the **problematic landscape at an affordable cost**.

## The Problematic Landscape

### Times are Changing

### Application Delivery

Times are changing. The application world is evolving in terms of how we service applications. It now requires a new approach to load balancing traffic. Applications are becoming more complex. What makes landscaping challenging these days is the connectivity of advanced gadgets like internal cameras, onboard computers, and many other wireless devices. Traditionally, we supported the monolithic single application per server design.

The single application per server was an apparent waste of resource triggering a variety of on-premise and cloud based multi-tiered applications. Each application stack has a number of tiers, potentially requiring load balancing functionality between each tier.

Microservices are also coming to this modern age. Microservices are fast and dynamic, individual services potentially located in different geographic areas posing additional challenges for load balancing and security. There is an increasing demand to support real-time interaction and multimedia content over a variety of network and device types.

### Application Availability

The average customer has high expectations and requires peak performance on all device types and networks. In this digital world, slow page loads are unacceptable and the customers expect services to work all the time.

### Increasing IoT Traffic Volumes

Traffic volumes are growing year by year. Gartner® predicts that The Internet of Things (IoT) is going to bring the number of connected devices to over 26 billion. Objects that were not necessarily connected to the Internet will now have IP reachability with the finesse to send and receive data with a variety of payload sizes.

IoT is all about data and is going to create a surge of data transfers through backend applications and the central IoT control panels will be empowered to handle the flow.

### DDoS Protection

Organizations are entering a new era of DDoS attacks. IoT not only increases the data quantity, but it also increases the DDoS landscape to billions of unsecured IoT devices. The network perimeter is no longer static, making it easier for cybercriminals to launch destructive DDoS attacks.

IoT Mirai BotNets are taking down the most respected networks with Terabyte scale DDoS attacks.

## Network Inconsistencies

### Heavier Web Pages

Technically, behind the scenes of a web page, a lot is happening. Web pages consist of many file types, scripts, and hundreds of objects; each object requiring an individual Hypertext Transfer Protocol (HTTP) request.

They become bulkier as the number of objects increase. As a result, the rendering of a page demands a more round-trip time (RTT), further degrading the user's experience.

### Different Network Types and Varying Latencies

Traditionally, web pages are designed for network links that exhibit slight packet loss and low latency, such as high speed fixed line and digital subscriber line (DSL) links. They often fail to meet the user's expectations to deliver adequate performance over lower performing network links that we have in the mobile phone world.

User connectivity over bandwidth constrained links combined with high latency and packet loss experience slower page loads, thereby inflating the response times for applications. There are varying latencies from the different network types. Fixed access connection such as DSL typically exhibits lower latency than wireless networks.

We also have plenty of bandwidth variations between these network types. Different networks with varying performance metrics have different results for web applications.

When mobile users access applications that are designed for faster-fixed line users, the results are unexpected. Much of this can be aided by a load balancer with an optimized TCP stack for both front and back end connections.

## Behavioral Impact

There are many studies available that demonstrate the behavioral impact of poorly performing applications. The ultimate goal for a web page is to load in less than 100 msec. After the 1 second mark, the user thought process is interrupted and after 10 seconds their dialogue gets changed. Therefore, the poorly performing applications translate to less revenue.

## TCP Insufficiencies

HTTP Web Applications sit on top of Transmission Control Protocol (TCP). However, TCP was developed in the 1980's and has many performance related inadequacies. It understands that applications exist on networks with no latency related problems. When TCP was created, the focus was on congestion control mechanisms, not latency.

However, the nature of HTTP makes it very sensitive to high latency. These days, almost all networks exhibit varying degrees of high latency. Latency varies widely, since so many factors contribute to it.

TCP has a lot of work to do under the covers, to provide reliable delivery. Unlike UDP, its guaranteed delivery requires a lot of overhead which degrades an application's performance.

TCP is the prime protocol in use today, and unfortunately you can't avoid this overhead if you want to communicate over the Internet.

## TCP Request/Receive Behavior

TCP does not support multiple parallel sessions. You need multiple TCP sessions for multiple sessions. It is essentially just a stream of bytes with no internal structure.

Its 3-way handshake exhibits significant delays even before the first user data is sent. TCP offers you a reliable stream that handles packet loss very well but it doesn't guarantee timely delivery. Therefore its orientation is not latency/performance focused.

## Latency is the Biggest Problem

Latency is the biggest problem with application performance. If you are expecting a high-speed web browsing experience, then the only available natural option is to shorten the RTT.

After a certain threshold increasing bandwidth does not decrease latency. We can buy bandwidth, but for latency, we need to shorten the cable. This is expensive unless we use a Content Delivery Network (CDN).

Nevertheless, a CDN cannot be used for everything. They are not so good with dynamic content and pose challenges in SSL certificate management.

### Cloud Load Balancers


To diversify the services, an increasing number of organizations are looking to place workloads in the cloud. The cloud is a scalable, location-independent and ready-made data center, allowing the enablement of multi-layer application architectures on demand.

Within multi-layer application architecture, load balancing services are commonly deployed between the Web, App, and Database tiers. Due to the request/receive nature of HTTP applications, the impact of a bad response substantially degrades the performance due to the quantity of how many requests are needed for a cloud web service.

An administrator can have one load balancer instance in front of all segments or multiple between segments, either Inline or one ARM Mode. Native cloud instances provide default load balancing but lack granularity of a full feature set required for advanced applications.

For this reason, load balancers with adequate feature sets are critical for cloud deployments and for the very functioning of cloud services.

## An Ideal Application World

### What is the ultimate goal?

The primary goal is to accelerate web applications across all networks and device types to improve the user experience with fast page load times. All this should come at an affordable price.

To do this we need to send data efficiently, infrequently, and as little as possible. Sending data efficiently can be done with optimizing both the front end and back end TCP connections. Infrequent data transfers are fulfilled with caching and compressing transmits to as little information as possible.

### The need for a new type of Load Balancer

Considering the rocketing traffic, vulnerability to DDoS attacks, and performance, it is crucial to choose a load balancer that is designed to outrun the shortcomings. The modern load balancer has been rolled out after extensive research and testing under current conditions. Therefore, it is certainly more in tune with the advances in application server software.

### What a Load Balancer should have?

What should the perfect load balancing solution offer? Traditionally, load balancing has often been considered an expensive appliance with high ongoing maintenance and consistent training cost.

First and foremost, a load balancer should be affordable. Contrarily, many load balancing vendors charge high on appliances along with high ongoing maintenance costs. They have made the feature set

so complex that consistent training is required on a yearly basis.

Load balancers offer a variety of features but it should only provide features matching with the customer's requirements and at a low cost. The majority of networks work perfectly fine with an active - passive mode opposed to trying to set up clusters with complicated active - active deployments.

A load balancer should offer a customized operating system with performance oriented TCP/IP stack, advanced HTTP Parsing engine with appropriate feature set. It should handle significant amounts of traffic along with the capability to support periodic bursts without degrading the performance.

Load balancers are in a perfect network position to provide adequate security, fulfill regulatory compliances, and protect against the increasing level of DDoS attacks.

### AVANU Product Set

| WebMux - Virtual Appliance | AVE-100 | AVE-300 | AVE-500 | AVE-1000 |
|---|---|---|---|---|
| Network Layers | 4-7 | 4-7 | 4-7 | 4-7 |
| O/S Processor Architecture (bit) | 64 | 64 | 64 | 64 |
| Internet Link Throughput (Max Gbits/s less Ethernet Overhead) | 1.0 | 3.0 | 5.0 | 10.0 |
| Servers/Farm Support (Max-Real/Virtual) | 4,999 | 4,999 | 4,999 | 4,999 |
| Technical Support | 1 Year | 1 Year | 1 Year | 1 Year |
| FIPS-2 Level 1 Compliant | Yes | Yes | Yes | Yes |
| TAA Compliant (Developed in USA) | Yes | Yes | Yes | Yes |

| WebMux - Network Hardware Appliance | A425 | A525 | A625 | A725 | A825 |
|---|---|---|---|---|---|
| Network Layers | 4-7 | 4-7 | 4-7 | 4-7 | 4-7 |
| O/S Processor Architecture (bit) | 64 | 64 | 64 | 64 | 64 |
| Internet Link Throughput (Max Gbits/s less Ethernet Overhead) | 4.0 | 4.0 | 40.0 | 50.0 | 80.0 |
| CPU Processor (# Cores) | Quad | 8 | 10 | 14 | 18 |
| ECC Memory (GB) | 8 | 16 | 32 | 64 | 128 |
| Network Type (Options) | Copper | Copper/Fiber | Copper/Fiber | Fiber | Fiber |
| Network Connector Type (Port) | RJ45 | RJ45/SPF+ | RJ45/SPF+ | SPF+ | SPF+ |
| IPMI Port | Yes | Yes | Yes | Yes | Yes |
| Management Port | Yes | Yes | Yes | Yes | Yes |
| Solid State Drive (SSD) | Yes | Yes | Yes | Yes | Yes |
| Smart Temperature Control Fans | Yes | Yes | Yes | Yes | Yes |
| Power Supply (Hot-Swap, 400w) | Single/Dual | Single/Dual | Dual | Dual | Dual |
| Servers/Farm Support (Max-Real/Virtual) | 4,999 | 4,999 | 4,999 | 4,999 | 4,999 |
| Front LCD Panel (Quick Configuration) | Yes | Yes | Yes | Yes | Yes |
| Chassis | 1U | 1U | 1U | 1U | 1U |
| Hardware Warranty | 2 Years | 2 Years | 2 Years | 2 Years | 2 Years |
| Technical Support | 1 Year | 1 Year | 1 Year | 1 Year | 1 Year |
| FIPS-2 Level 2 Compliant | Yes | Yes | Yes | Yes | Yes |
| TAA Compliant (Developed & Manufactured in USA) | Yes | Yes | Yes | Yes | Yes |

AVANU has virtualized load balancer versions in Virtual Machine (VM ) format. It can be started on any cloud to take advantage of all cloud benefits along with a rich feature set. It is supported on all cloud platforms and major hypervisor models.

## Hardware Components

AVANU platforms comprise of off the shelf hardware parts from reliable and reputable manufacturers with large fleet sizes. They have selected newer component technology and optimized the design to fit everything into a lean 1U highly available device.

Smart partitioning of the SSD drive and dual hot swapping power supplies along with an IPMI port that give customers the reassurance that every step has been taken to mitigate hardware failure and protect the device uptime.

## Solid State Drives

The latest in Solid State design for WebMux eliminates hard drive failure for enhanced reliability and high availability. The SSD is used for the O/S and log files. It consists of a two partition design useful to mitigate any upgrade failures.

In keeping up with the latest technology, AVANU has moved away from the older SSD storage design to a current, contemporary design. AVANU employs the latest in Solid State design that speeds up the storage functionality with PCI Express lanes. The company has now fabricated the possibility to use up to 4 lanes.

The Solid State approach enables parallel processing of the storage read and write requests, providing up to 65,536 command queues with up the 65,536 commands per queue. It's compact and plugs in directly into the PCI bus, eliminating the need for additional cables and reducing the overall storage size requirement while increasing reliability.

The design dramatically cuts down on assembly times giving AVANU the competitive advantage of a quicktime supply to market. All the parts are assembled in-house and within the USA that gives an upper edge to this 'latest in SSD design' company.

## DDR4 ECC RAM

AVANU employs high density DDR RAM categorized as server grade memory for high performance and reliability.

Error correcting code or "parity" memory can auto-correct simple memory errors on the fly. The chip count is evenly divisible by 3 or 5 and the additional extra chip is used to make sure that data was processed correctly.

## Built-in Hardware AES Acceleration

The Central Processing Unit (CPU) uses an Advanced Encryption Standard Instruction Set that boosts the performance of the applications performing encryption and decryption. AES is a block cipher which means a string of plain text is chopped into blocks. AES uses 128 bits per block, twice the size of DES.

It uses a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. AES is a widely-deployed encryption standard and the WebMux CPU is used to accelerate AES encryption even further.

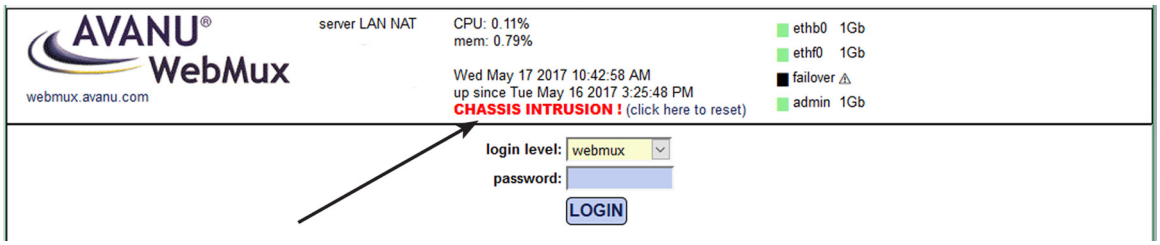
AES functionality involves implementing sub-steps of the AES algorithm into the hardware, thereby dramatically improving the encryption, decryption, and key generation.

Four instructions relate to the AES encryption and decryption functions and two other instructions for AES key generation. The sequence of these instructions creates a full encrypted block. The instructions resides in the CPU and since this materializes in the hardware, WebMux benefits from the performance gains.

For example, if you have 12 threads, you can have 12 different instances of using these extended introductions for encryption and decryption occurring at the same time without interfering with each other.

## An Intrusion Detection

AVANU employs FIPS (Federal Information Processing Standards-cryptographic modules) by adding a physical toggle micro switch to the chassis offering an open/close intrusion notification. Upon physical tampering, the device signals to the hardware registers.




## Web GUI

AVANU has created a responsive Web GUI that magnifies the user's experience while administering the WebMux.

Automatic device detection allows the WebMux to resize and rearrange the GUI according to device type.

Desktops consist of a landscape layout while a mobile device will have a portrait layout. For ease of use, the GUI is designed to push the most important menus to the top of the screen.

The GUI works the same on small and large screens. Mobile devices do not have the concept of mouse events. The screens behave different depending on its size and what is showing where on the screen. The software package for the GUI is optimized to impart a similar experience from all device types.

## LCD Screen

The LCD screen on the front panel gives you direct configuration access from the device itself. It enables you to jump start the WebMux configuration without needing anything more than the WebMux itself.


It acts as an alternative path to device configuration if for some reason an administrator cannot connect to the GUI. Therefore, it offers a redundant way of getting the machine's attention.

# AVANU Load Balancing Functionality

## Local Traffic Managers

AVANU load balancers perform as local traffic managers offering more than just a network-centric viewpoint. Residing outside of the application back end servers they can load balance with a variety of networking techniques rather than administrators digging deep into the code of the application.

There is a clear distinction between the physical server and the application residing on the server, further allowing a single physical to host a variety of applications.

## TCP Connection Management

Most of the time, the load balancer acts as a full TCP proxy sitting in the middle of two distinct TCP connections: client and server side connections. The server terminates the TCP sessions which enables it to carry out any client side and server side TCP enhancements.

At the front end we can carry out TCP optimization technique to make use of entire available bandwidth. At the back end, techniques such as TCP multiplexing are employed to further improve application performance.

Load balancers act as full proxies enabling you to inspect, encrypt, or decrypt all traffic. It traverses to your network enabling intelligent traffic management. They act in a unique position to make every connecting client and server connection work more efficiently; improving application performance and user experience.

# AVANU Load Balancing Topology Modes

All AVANU models feature the following load balancing topologies:

1. NAT Mode
2. Transparent Mode
3. Single Network Mode
4. Direct Server Return Mode

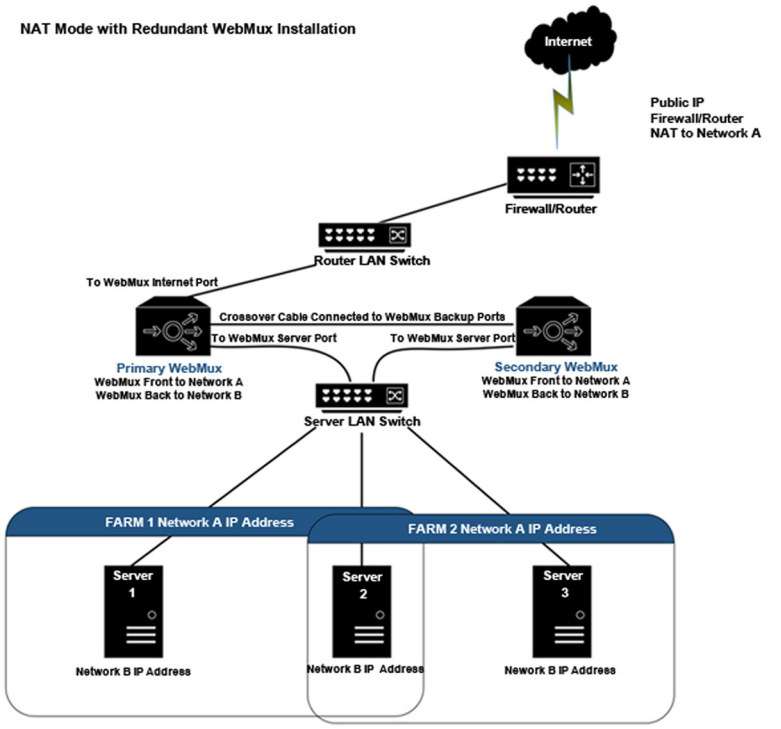
The choice of topology is usually depicted in what the environment is currently in place or planned. If the farm virtual IP address is on a different network from the servers then choose the following NAT mode.

## NAT mode

Destination NAT is performed between the main network and the segment containing the servers. The original client IP is preserved useful for logging purposes.

The WebMux behaves like a smart router performing load balancing algorithms sitting between two different subnets. The server and farm IP are on different networks.

The WebMux must complete the transaction between the original client and server; return traffic must flow through it. As a result it could involve network changes on the server side.




It is significantly useful for scenarios where you want the load balancing device to perform like a Firewall. NAT mode provides security for isolating servers from other parts of the network. The WebMux can firewall all uncofigured ports, adding another layer of security.

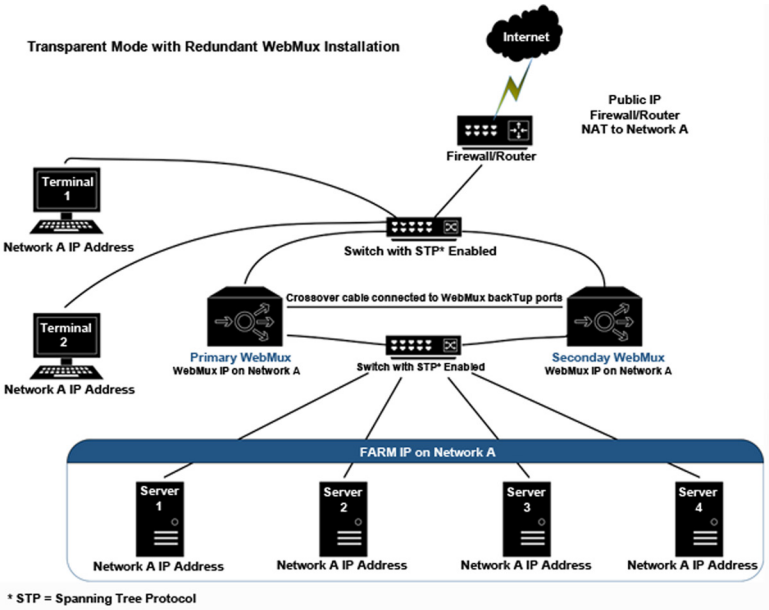
## Transparent Mode

With Transparent mode, no NAT is performed. The original client IP is preserved. The WebMux acts similar to an Ethernet Bridge and sits between two physical segments creating one single logical segment.

In Transparent mode, the servers reside “behind” the WebMux but are effectively in the same network as the server “in front of” the WebMux.

Any traffic related to load balancing is load balanced, any traffic not related to load balance can go through WebMux like a piece of network cable - transparently.

All servers can have an external IP address, whereas in NAT mode they must have internal IP addresses. Transparent mode does not offer firewall protection. All servers can talk to each other freely across the WebMux.



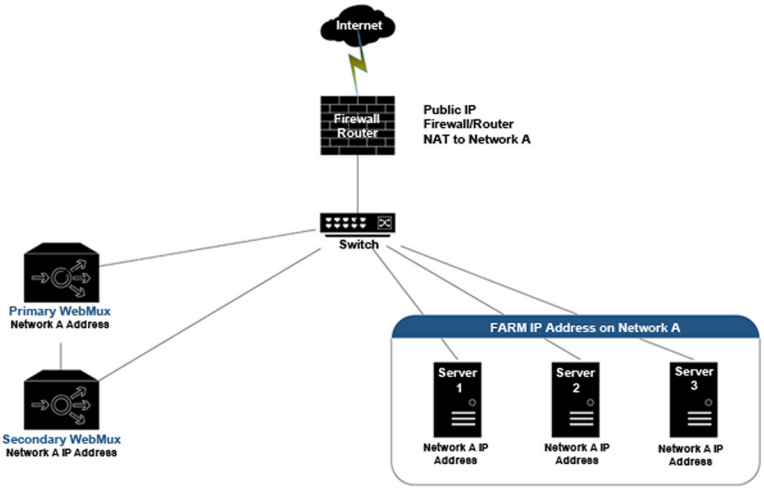

One of its main benefits is the ease of deployment without the need to change any server or gateway IP

addresses. It is useful for scenarios where you cannot make any logical network changes.

## Single Network Mode (Plug and Play)

Source NAT is performed with Single Network Mode. The original client IP is replaced but additional MIME Header is used for client IP preservation. Server and farm IP are on the same network.

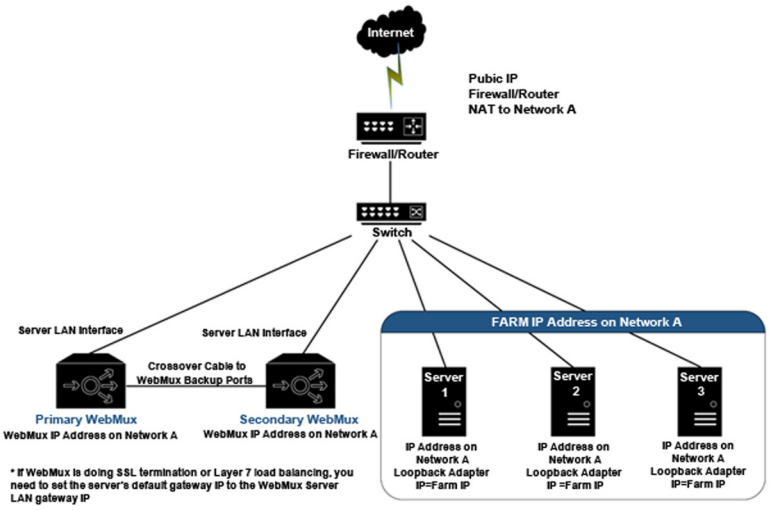
The WebMux acts like a proxy and all traffic appears to come from it. Physical port bonding is available for additional throughput. Servers remain unmodified and can still use the existing gateway; return traffic will naturally flow via the WebMux as the server see the WebMux as the one making the request.




In this mode, the firewall protection gets disabled from the WebMux. All servers can talk to each other freely across the WebMux. Load balancing occurs when the farm IP is accessed. Single Network Mode is useful for plug and play scenarios.

## Direct Server Return Mode

Direct Server Return Mode does not perform any NAT. The original client IP is preserved. The server and farm IP are on the same network. Similar to the Transparent mode, the servers can have external IP addresses.




The WebMux carries out packet forwarding using media access control (MAC) address of the servers. Operating system independent modifications are required on the server side so they do not respond to Address Resolution Protocol (ARP).

Servers remain unmodified to use the existing gateway to bypass the WebMux and respond directly to clients. SSL terminates the traffic on the WebMux and the Layer 7 load balanced traffic must continue to use the WebMux for return traffic.

This mode is useful for scenarios where applications have large payloads such as Big Data deployments. Direct Server Return Mode offers higher performance for server return traffic since it does not need to pass through the WebMux that potentially under certain designs turns into a bottleneck jam.

An important aspect to efficient load balancing is the decision on how to distribute the load among backend servers. Scheduling is carried out using a number of metrics. Metrics are session based (not per frame) as all frames in a UDP or TCP must be forwarded to the same server as to not break the session.

AVANU employs both basic and advanced algorithms to determine how application traffic is distributed to servers. Scheduling is a farm level setting that affects servers that are members of that farm. The scheduling method determines what algorithms are used when load balancing and traffic managing servers are within a farm.

# AVANU Scheduling Methods

AVANU solution consists of three core scheduling methods, each with optional variations for further granularity.

1. Least Connections
2. Round Robin
3. Fastest Response

## Least Connections

The WebMux prioritizes sending clients to servers with the least amount of connections at any given time. The distribution may not be even as clients potentially stay connected to a server for a longer period of time.

## Round Robin

The simplest form of scheduling is round-robin. The load balancer allocates connection based on a list and assumes that all connections will have a similar load. The WebMux sends clients to whatever server is next in the list using a round robin fashion.

Server resource allocation is not considered; the scheduling methods assume that all the servers can handle the same load level. Over a longer observation time, the round robin scheduling method has an even distribution.

## Fastest Response

Faster Response takes into account the last connection that the WebMux sent a client to a server. Real servers with faster responses are favored for further client connections.

The results are dynamic and will change the preferences accordingly. Each of the three core algorithms is further granulated with additional weighted and persistence options; used together or separately.

## Weighted Variations

The Weighted Variations involve setting priorities manually to individual servers with weighted values. It is useful in scenarios when individual servers have additional capabilities of handling extra client connections.

However, if all the server's load-balances are of equal capacity then the weighted option would be unnecessary. When you add Weighted Variations to the load balancing scheduling, it shifts the algorithms slightly; changing the expected behavior.

## Persistence Variations

The Persistence Variation ignores the scheduling algorithm and places clients on the same server upon reconnection. If the same client returns to the server farm and reconnects, the WebMux remembers parameters of the client and if a certain amount of time has not elapsed, the client is placed on the same server.

The WebMux specifies a timeout period determining how long a client can disconnect and reconnect and be sent to the same server in that farm. Outside the period, the Persistence variation is not valid and connectivity falls back to the configured scheduling algorithm.

Persistence is useful for solutions when you are unable to track sessions among the servers in the cluster. The load balancer provides persistence unless the backend server has tracking capabilities, for example through Active Directory (AD) or (Lightweight Directory Access Protocol) LDAP synchronization.

Persistence is often used in shopping cart scenarios where there may be instances of long periods of time where the client is inactive and needs persistence provided by the load balancer to fulfil the online transaction.

Similar to the Fastest Response scheduling algorithm, Persistence Variations may offer uneven distribution.

## Persistence Variations - Layer 4

For Persistence with Layer 4 load balancing, the source IP address and the connecting server are recorded for future sessions. The easiest way to enable persistence is on the source IP address. However, it becomes more complicated with advanced network designs, such as the user's sitting behind a proxy.

In these scenarios, administrators must move up the stack and use other more advanced criteria for session persistence.

## Persistence Variations - Layer 7

Persistence with Layer 7 load balancing can be done in a number of ways - matching patterns in the URI (Uniform Resource Identifier) in the client's GET request Header, Host MIME header or by issuing and tracking a Cookie.

A Cookie is a piece of information used to identify a client for future requests; either permanent or temporary. Layer 7 HTTP Cookie Load Directing tests the match pattern against the Cookie MIME Header contents.

A Cookie is issued to a client upon connecting to a particular server. The WebMux will generate its Cookie to keep a track of which client session belongs to which server.

If the client disconnects and the WebMux detects the same Cookie upon reconnection, the connection

will automatically be sent to the same server. This is useful for shopping cart services, for example, so that the client will be directed to the same server and keep their shopping cart items valid.

Layer 7 HTTP Virtual Host Load Directing with Cookies allows you to direct traffic to the name-based virtual hosts. This scheduling method allows you to have several name based virtual hosts on a single physical server with one IP address.

# AVANU Health Checks

## Application Specific Health Checks

Load balancers prudently ensure the reliability and availability by monitoring the status of backend applications so that requests are successfully answered. Health checks determine the best server to service the inbound request, ensuring that the application will always be served from a healthy node.

There are multiple levels of health monitoring and the granularity intensifies with the level, depending on the application complexity. The type of check varies depending on the type of hosted application and the required complexity.

The most basic would be a simple PING testing network layer reachability. Or at a more advanced level, the load balancer can inspect packet headers for specific keywords or request certain file types.

More complex health checks are required when the network stack is functioning but the problem lies somewhere higher up the stack, in the application layer. AVANU health checks are application specific. The WebMux examines server responses to validate whether it's operational or not.

The WebMux health checks suit all application requirements - custom created scripts, HTTP Checks, TCP connects, DNS, NTP, POP3, SNMP, and a basic level PING.

An HTTP health checker would examine the HTTP returns code to validate functionality. An HTTP 200 or 404 would deem that the HTTP service is fully operational. For additional granularity, the return codes are further tailored to specify what response codes are valid.

Generic TCP and UDP checks offer basic initial connectivity checking. They don't go deep into the applications and simply look at the connectivity status. Custom scripting health checks are the most sophisticated and offer a lot of flexibility to meet non-standard health checking requirements.

AVANU custom scripting leaves it wide open for customers to create productive performance health checks. They can either come in the form of a Shell Script or C program.

They are useful when you want to check resources on the server that are not directly reported by the application itself. For example, a custom script is crafted to ensure that the CPU and RAM are at a certain level while breaking a predefined threshold or it would be considered an unhealthy server.

Scripted health checks are also useful to test remote servers, for example, a database holding structured data used by servers.

# AVANU Load Balancing Features

## Layer 4 Load Balancing

Load balancing generally falls into two buckets - both Layer 4 and Layer 7. Layer 4 load balancing is the most basic and common way to load share servers. Most of the time, it distributes the load according to the source IP address of the client.

Layer 4 can distribute the load according to Layer 2 - Layer 4 information such as MAC/IP address and TCP Port number. More than often, they act upon data found in the Network and transport layer protocols. Layer 4 operates at a known point within the data packet that never changes.

## Layer 7 Load Balancing

Application architectures evolved, so did the need to examine past TCP port numbers. Layer 7 load balancing examines varying values within the content. It analyses information generated by the application itself.

Layer 7 goes deeper into the application layer data attributes in order to make decisions. It may examine, for example, SSL session ID, Cookies, MIME headers, or other HTTP Header attributes. They act upon data found in Application layer protocols.

Supply the WebMux with a regular expression allowing the filtering and then match the incoming connection according to URI, Host MIME Header or Cookie. If the WebMux finds a matching set, it will send those clients to the server that match the regular expression.

### Layer 7 HTTP URI Load Direction

Layer 7 HTTP URI Load Direction directs traffic depending on a match pattern tested against the URI in the client's GET request header. HTTP URI Load Direction is commonly used for hosting a group of servers serving a variety of content. One group of Media servers is serving Media files with a specific URI and another group of Media servers is serving HTML files with a different set of URI.

HTTP URI Load Direction enables the WebMux to direct to the servers according to URI enabling administrators. This will compartmentalize which servers you want to use for specific resources.

## Cookie Load Direction

Instead of looking at the URI, Cookie Load Direction examines the Cookie MIME Header in the request. Cookie Load Direction is useful in scenarios when you have a cluster of servers and you only want one of those servers to handle the clients that are not logged in.

## SSL Offload

SSL transactions need to be encrypted and decrypted, resulting in a drag on performance. AVANU operates with the latest SSL enhancements and uses SSL Acceleration for faster cryptographic processing, thus better performance.

SSL encrypts TCP applications by encrypting the data portion of the session only to leave the IP/TCP and SSL headers clear for session persistence. The WebMux supports TLS v1.0, TLS v1.1, and TLS

v1.2 with RSA key length from 1024, 2048, 4096, and 8192 bits. For each WebMux, one can have 32 SSL
certificates.

All the HTTPS incoming traffic will be sent or terminated to the farms on HTTP port (80). After the WebMux terminates the SSL traffic, only clear traffic will go to the servers.

For server return traffic, the WebMux will re-encrypt the data. The encrypted data will be sent back to the client instead of the back end servers performing the SSL. The WebMux performs the SSL negotiations between the clients and servers and performs the SSL processing.

One of the main advantages for the load balancer to terminate the SSL is for management purposes. You only need to update one Certificate to update the entire cluster of servers.

Because traffic between the WebMux to the back end servers is unencrypted, your servers will not be able to determine if the originating connection was HTTP or HTTPS. To validate this type of information you have the options to “tag SSL-terminated HTTP requests” adding a MIME Header “X-WebMux SSL-termination: true”

The WebMux can terminate the SSL session and send either of the encrypted traffic to the real servers. A fully encrypted path is useful in high-security situations where every path and node must be encrypted.

One can also block non-encrypted incoming traffic so that only encrypted traffic can reach your server. This might be useful, when you only want encrypted traffic to reach your servers.

## HTTP Compression

HTTP Compression reduces the amount of data accelerating application transactions, thereby improving transfer speed and bandwidth utilization. If the client’s web browser sends out a MIME Header it means that it accepts compressed data. The WebMux will compress the HTTP data to the client browser.

If the WebMux detects that the server in the farm is already compressing the data, it will not perform the compression. Instead, it will let the compressed data from the server pass through without processing.

## HTTP Caching

Caching is useful if you have static web pages. For static web pages, the requests to the server are not required every time a request is carried out. Caching is an essential ingredient for fast websites. Caching is primarily a collection of items stored for future use.

Asset management for HTTP caching may include static assets (example - images and CSS files) as these files don’t or rarely change. Caching improves the performance of the website by temporarily storing static data that was recently accessed.

Administrators can have their WebMux cache static pages and the WebMux acts as the server to serve the pages. Requests are made to the WebMux instead of making the query to the server behind it.

Traditionally, caching is often deployed for outbound connectivity whereby client’s cache targeted web pages. AVANU has exceptionally reformed the mechanism to cache the incoming requests and

load on the backend servers.

## DDoS Protection

DDoS is a growing concern and even the lightest of the features can deter a DDoS attack from bringing down critical resources. The WebMux series support both Flood Contro® and Automatic Attack Detection.


The WebMux Flood Control feature was specifically developed for a United States service organization of the Department of Defense in 2012. The feature is very effective for stopping DDoS attacks at an IP/UDP/TCP/ICMP level. Most volumetric attacks operate at this layer of the Open Systems Interconnection model (OSI) Model.

Flood Control relates to the packet/rate level from any IP addresses. If a source IP is generating too many requests, it will be blocked. Automatic Attack Detection is another effective DDoS mitigation feature. It allows administrators to set a limit on maximum amount of concurrent connections permitted from a single IP address.

## Multiple Address/Port (MAP)

The WebMux load balancer has Multiple Address/Port capabilities supporting intelligent failover of rich media applications. The WebMux can handle multiple external IP addresses and individual server can be shared across multiple IP addresses.

The WebMux supports multiple ports to be logically bound for failover purposes, thereby providing automatic redirection of all related ports in a failover event.

## FIPS 140-2

The WebMux has firmware that supports the government required FIPS 140-2 standard. The FIPS 140-2 standard specifies the security requirements that are met by a cryptographic module utilized within a security system protecting sensitive, but unclassified information.

The WebMux series use the latest industry-standard SSL performance enhancements according to NIST certification guidelines. The SSL FIPS library provides confidentiality, integrity, and message digest services.

AVANU SSL FIPS natively supports the listed algorithms: DES, Triple DES, AES, RSA (for digital signatures), D-H, DSA, SHA-1 and SHA-2. SSL FIPS performs ANSI X9.31 compliant pseudo-random number generation.

Together with the WebMux built in electronic physical intrusion detection, AVANU follows NIST's standard guidelines for FIPS 140-2 Level 2 compliance.

# Summary

AVANU®
Discover the Difference

## WebMux™
### Network Traffic Manager

**Enterprise-class Server Load Balancing Solution**


**Quick to Deploy • Easy to Manage • Reliable High Performance • Fantastic Value • Affordable**

Load balancers are ideal to execute the organization's plan for end-user performance, reliability, availability, and scalability. AVANU offers the best of both worlds, lean load balancing at a low cost enabling business continuity and improving overall system performance. This enables the organizations to meet the most stringent Service Level Agreement (SLA) for the most complex Web application stacks.

AVANU has compiled all the best elements of a load balancer into a complete traffic management solution. Anyone can compile a load balancer but what matters most is the support and convenience of the entire solution.

AVANU experienced professional service team concludes that they are the market leader for lean load balancing at a low cost. Therefore, may it be in terms of performance or user-friendly execution or the cost effect, AVANU load balancer range offers an unmatched solution to your needs.