Deployment Guide
May 2016 rev. a

Deploying APV Series Application Delivery Controllers with Blackboard

1 Introduction

This deployment guide provides an overview of configuring the APV/vAPV application delivery controller for Blackboard applications.

Blackboard is a leading virtual learning environment and course management system developed by Blackboard Inc. It is widely used as a learning tool among K-12, colleges and universities, and is also used in large institutions and businesses. It is Web-based server software that features course management, a customizable open architecture, and a scalable design that allows integration with student information systems and authentication protocols.

Array Networks' APV Series application delivery controllers provide Layer 7 application load balancing, SSL offloading, Web security, Web compression and caching, header insertion and extensive usage reporting capabilities that are needed to keep applications running in their power band, even during heavy workload periods such as the beginning of a semester and enrollment process .

2 Prerequisites and Assumptions

Blackboard

This document is written with the assumption that you are familiar with the Blackboard solution. For more information on planning and deploying the Blackboard solution, please reference the appropriate document at library.blackboard.com

Array Networks APV Series

The APV/vAPV appliance must be running version ArrayOS TM 8.x or later. For more information on deploying the APV/vAPV appliance please refer to the ArrayOS TM Web UI Guide which is included in the product Web User interface or at the Array Support Portal website. We assume that the APV appliance is already installed in the network with Management IP, interface IP, VLANs and default gateway configured.

3 APV Application Delivery Controller (ADC) Benefits

The Array Networks APV Series application delivery controller delivers all required application delivery functions for optimizing application delivery for Blackboard enterprise environments, such as Layer 3 to Layer 7 server load balancing, high availability, SSL acceleration and offloading, DDoS protection, TCP connection multiplexing and failover - all in a single, easy-to-manage appliance.

Availability & Scalability

The APV's server load balancing capability ensures maximum uptimes and load distribution to scale Blackboard environments to meet capacity and performance needs.

SSL Offloading and SSL Security

The APV Series offers industry-leading performance and lowest cost per SSL TPS for 2048-bit SSL, along with advanced client certificate handling for secure application support and easy application integration. SSL acceleration reduces the number of servers required for secure applications, improves server efficiency and dramatically improves application performance. Offloading compute-intensive key exchange and bulk encryption, and delivering industry-leading client-certificate performance, SSL acceleration/offloading is ideal for scaling business-critical applications such as Blackboard that require high-volume secure connectivity.

Network and Server Protection

The APV appliance can protect Blackboard services from malicious network and server attacks like DDoS attacks, SYN floods, TCP port scans, UDP floods and UDP port scans, etc. The advanced rate limiting options can rate limit connections per user and advanced HTTP profiles can limit http commands and parameters for Web applications.

Site Resilience

The APV's server load balancing directs traffic away from failed servers and intelligently distributes services between servers based on capacity, load and response times for maximum performance and availability.

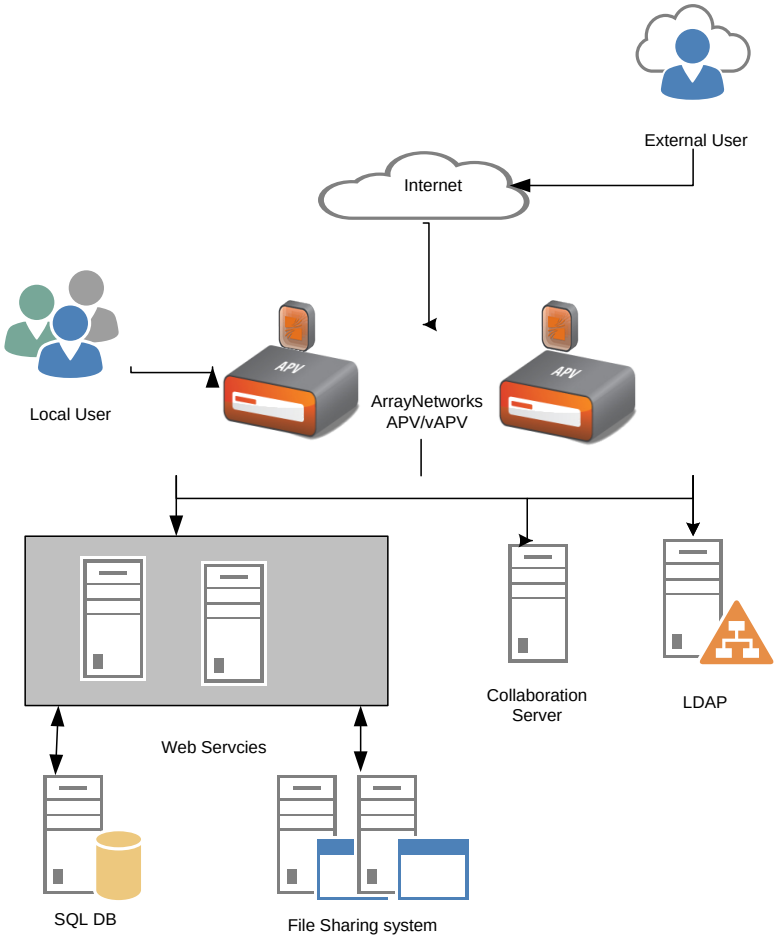
TCP Connection Multiplexing

The APV appliance multiplexes several client TCP connections into fewer connections for HTTP based services. The APV appliance also reuses existing server connections.

Cache offload

The APV appliance serves frequently requested content from cache for increased performance, helping scale capacity for Web-based services.

4 Configuration Scenarios



4.1 Deployment Considerations

Array Networks' APV/vAPV provides three scenarios typically used in Blackboard deployments. Following are the three most common high-level TLS configurations:

- HTTPS/TLS Offloading is where the Load Balancer communicates with the client using TLS but decrypts the sessions and communicates with the Blackboard application servers using HTTP. (For Blackboard Learn 9.1 April 2014 and later, TLS offloading is not optional. Full TLS is enabled by default. Hence this is not the recommended option for 9.1+ versions)
- HTTPS/TLS Re-encryption is where the Load Balancer communicates with the client using TLS (HTTPS), decrypts the sessions so that it can read the payload (cookies etc.), and then re-encrypts the session and communicates with the Blackboard application servers using TLS (HTTPS). This setup is most recommended to provide session-based load balancing and to inspect the traffic for security rules. Wildcard or SNI-based certificates can make the SSL termination feature very easy and cost-effective with a full Layer 7 protection stack and logging of all user requests.
- HTTPS/TLS Pass-through is where the Load Balancer communicates with the client using TLS (HTTPS) but does not decrypt the TLS session at all and just passes the

session on to the Blackboard application servers. Because the Load Balancer cannot read the payload, it has no access to cookies; it can only persist sessions to the application servers using IP-based persistence. This setup may not work well for requests coming from mega proxy or Web proxy systems. This is the only solution possible when the load balancer does not have access to the SSL key for importing into the APV for SSL decryption and re-encryption.

4.2 Configure the APV/vAPV Device with HTTPS/TLS Offloading for Blackboard

This scenario is a basic Blackboard server deployment which places the APV/vAPV in the middle between users and Blackboard Web servers.

Application/ Service	Virtual Service		Real Service		Health Check
	Protocol	Port	Protocol	Port	
Blackboard	HTTPS	443	HTTP	80	HTTP

4.3 Configure the APV/vAPV Device with HTTPS/TLS Re-encryption for Blackboard

In this scenario, the APV/vAPV system is a reverse proxy. The system is placed in the network between the clients and the servers. It provides secured, scalable, and highly available server offload and is completely transparent to the application users

Application/ Service	Virtual Service		Real Service		Health Check
	Protocol	Port	Protocol	Port	
Blackboard	HTTPS	443	HTTPS	443	HTTPS

4.4 Configure the APV/vAPV Device with HTTPS/TLS Pass-through for Blackboard

In this scenario, the APV/vAPV system is a reverse proxy. The system is placed in the network between the clients and the servers. It provides secure, scalable, and highly available TCP offloading with IP-based access to the Application Users.

Application/ Service	Virtual Service		Real Service		Health Check
	Protocol	Port	Protocol	Port	
Blackboard	TCP	443	TCP	443	HTTPS

5 Configuring APV/vAPV for Blackboard Services

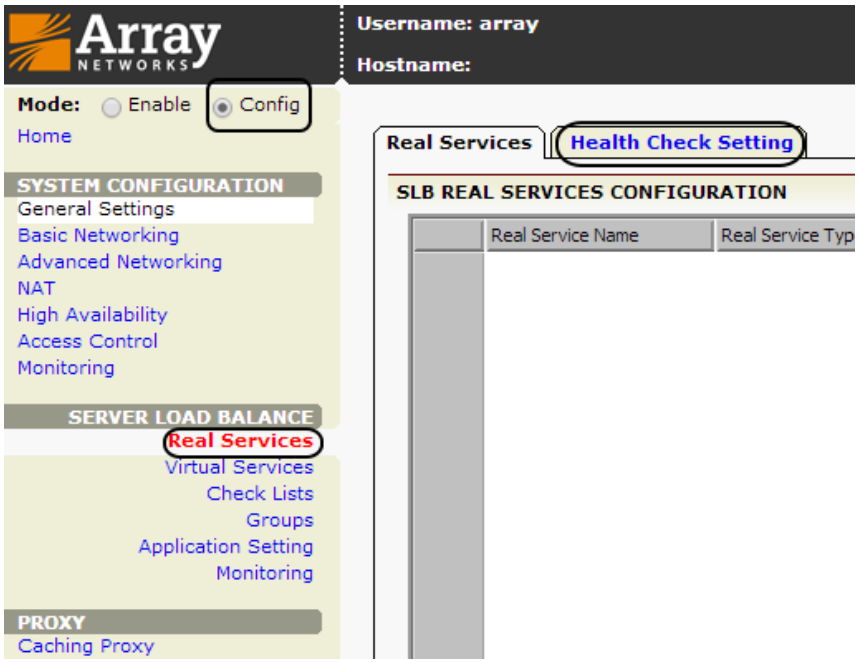
5.1 Configuring APV/vAPV for Blackboard Users

This section assumes that Blackboard servers are running HTTPS-based Web services for their applications which is scenario 4.3 in the above discussion.

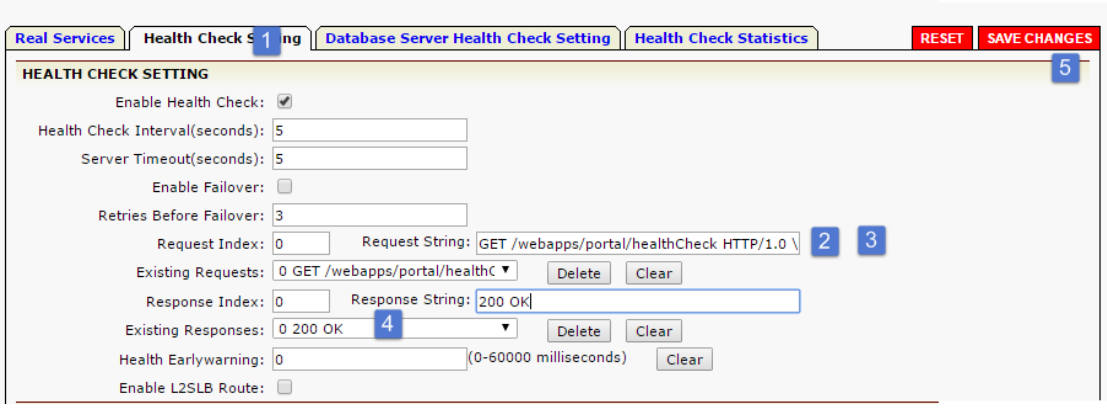
5.1.1 Create a Blackboard Health Check

Make certain you are in Config mode and have selected the feature Real Services from the sidebar. The configuration window will display two tabs, Real Services and Health Check Setting.

For a better application service Health Check, a simple HTTP content health check can be better than a TCP/ICMP health check for service availability:



1. Click on the "**Health Check Setting**" tab, a new window will display.
2. Input Request String (without quotes) "**GET /webapps/portal/healthCheck HTTP/1.0 \r\n\r\n**" [see figure below].
3. In our example we need to input "**GET webapps/portal/healthCheck HTTP/1.0\r\n\r\n**" in the Existing Requests field
4. Input "**200 OK**" in Response String
5. Finish the Health Check Setting by clicking "**SAVE CHANGES**"



5.1.2 Create a Real Service

Add the Blackboard Web servers in the real server profile with the associated health check. Add each server with its name, IP/port and protocol information as an APV SLB Real Service using the following steps. Please ensure the server health check is up and green (for active status) after this configuration.

1. Select the action link **"Add Real Service Entry"**. The configuration window will present a new screen for **SLB REAL SERVICES CONFIGURATION**.

2. **Enable this Service**: Check Box

This check box enables or disables the Real Service. If disabled, APV will not dispatch new traffic to the Real Service.

3. Input the Real Service Name; in our example we input "**RS_BB01**" as the first Real Service name. (No spaces are allowed for names)

4. Select **HTTPS** as the Real Service Type.

5. Input the Blackboard Web Server IP "**10.1.1.72**" and **443** as the Real Service Port.

6. Connection Limit: **1000**

This sets the maximum connections to the real service. This setting helps with application stability without overloading the server or application. Increase the number if the server is capable of handling greater loads.

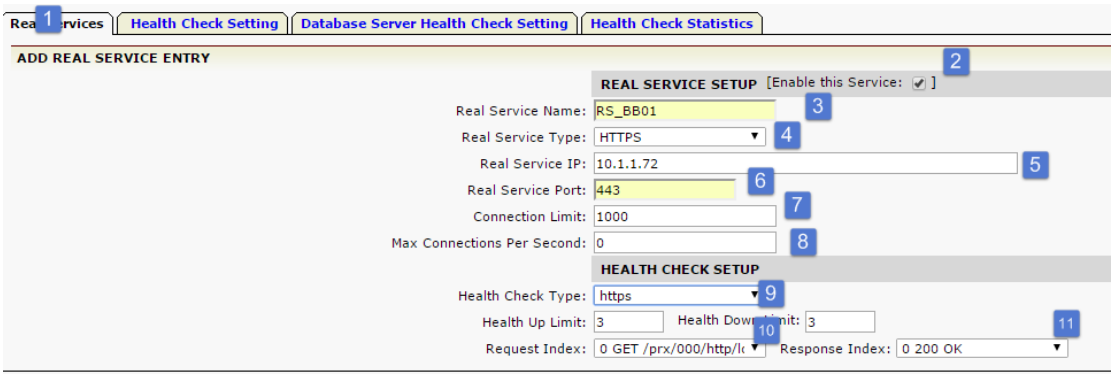
7. Max Connections Per Second – leave as default **0**. If the Real Server application has a performance issue, the APV's SLB capability allows connection rate limiting to the backend service.

8. Select **HTTPS** as the Health Check Type.

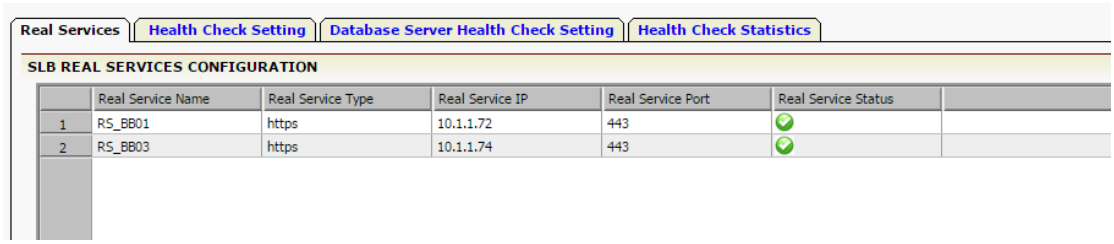
9. Select "**GET /webapps/portal/healthCheck HTTP/1.0 \r\n\r\n**" as the Request Index.

10. Select Request Index **0,** which has "**200 OK"** as the Response Index.

We expect the Blackboard server to return an HTTP status code 200, and an unsuccessful response would usually be 4xx or 5xx for the HTTP status code.



1. Depending how many Blackboard Web servers there are in your environment, you can click "**Save and add another**" real service (i.e. another Blackboard Web Server) using the same procedure. You can see the real service status when you have finished creating Real Services.

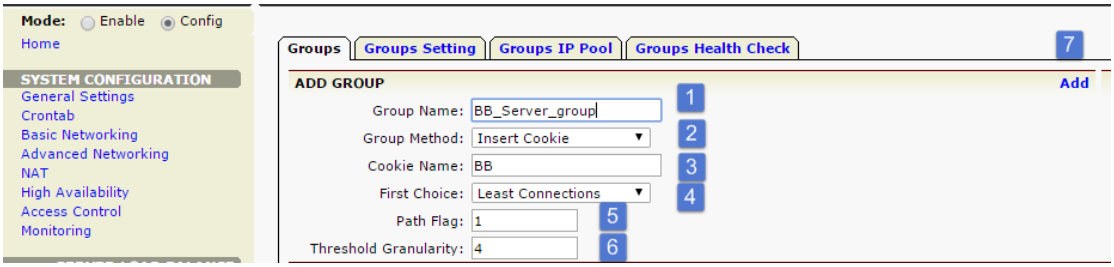


5.1.3 Create a Service Group

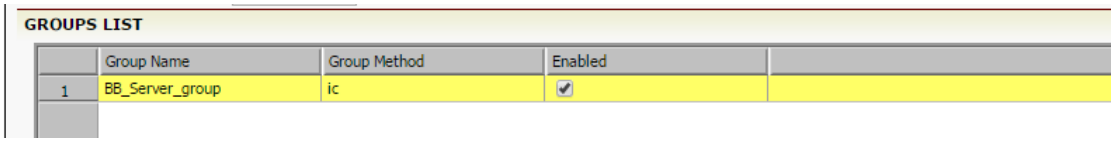
Make certain you are in Config mode and select "**Groups**". The configuration window will display two tabs, Groups and Groups Setting. Select the group policy **Insert Cookie** with the appropriate server priority and weight. Assuming both the servers have the same weight and priority, the APV will have the configuration below.

1. Add Group Name "BB_Server_group" as below.
2. The Group Method is Insert Cookie which will insert a server-specific cookie per session for tracking sessions and persistency for users
3. The cookie name could be anything to denote the cookie header that will be inserted by APV device which exists only between the browser and the load balancer. Here the sample cookie name is BB
4. Select Least Connection as First Choice
5. Path Flag is 1 (indicates the same as the full domain)
6. Threshold Granularity is default 10; 4 is a good value for Blackboard services to ensure a similar load for all servers in the group

The Insert Cookie Group Method sets a cookie name=Real Service Name to allow tracking of user persistency on each server. The cookie is only used between the user and the load balancer, and is not passed to the server. Insert Cookie as the Group Method, with Least Connections as the First Choice, is the preferred option for load balancing traffic between different Blackboard services.

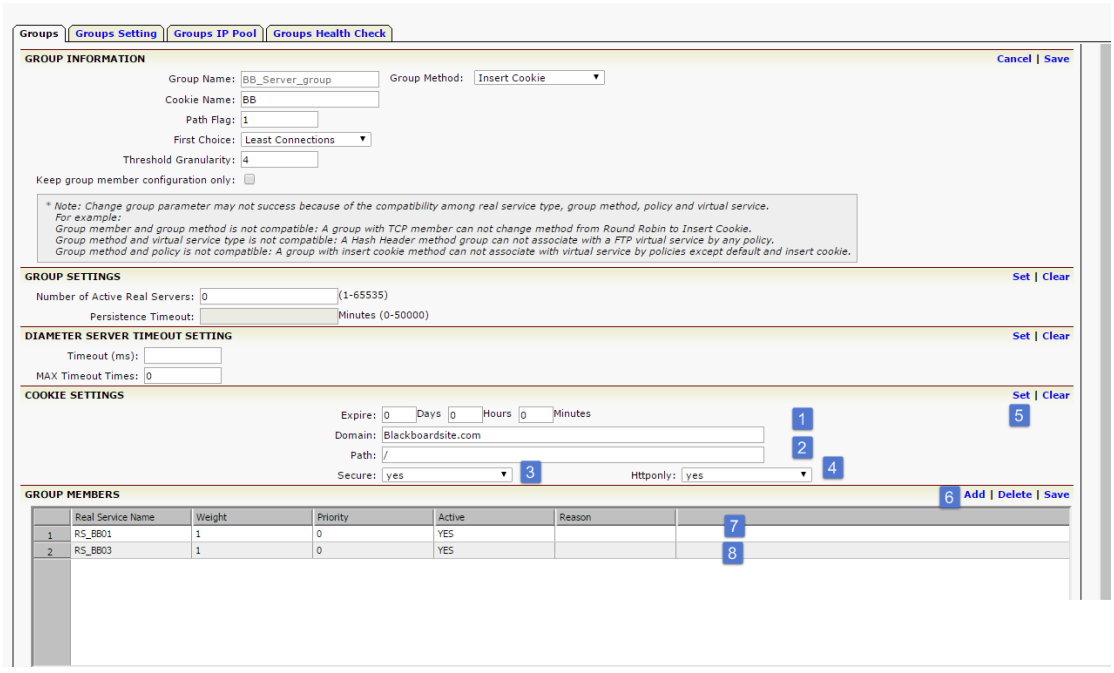


Click ‘Add’ at the top right of the screen to add your new Group. The following screen will appear. Double click the Group.



Enter the details for the secure cookie parameters for domain, path, secure and HTTP-only cookie.

Add the servers as Group Members with the appropriate priority and weight attached to each server.



5.1.4 Create a Virtual Service

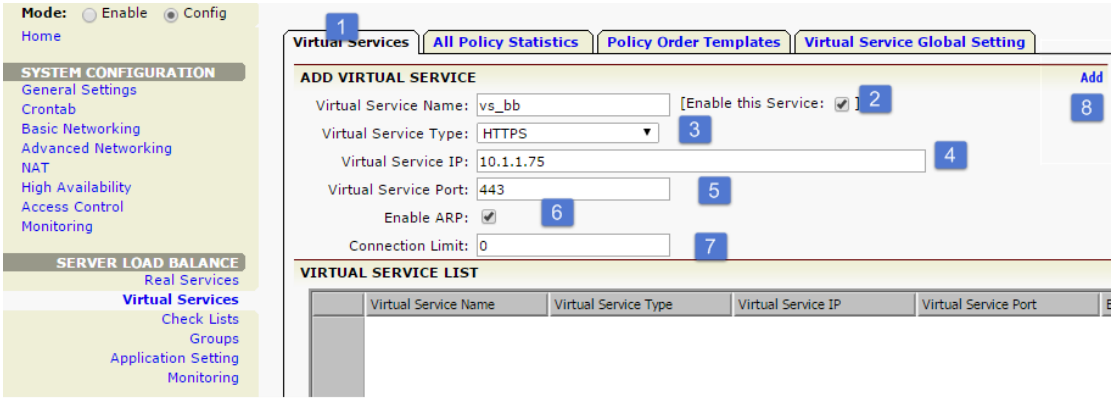
The next step is to create an APV SLB Virtual Service for clients to access these services. On the APV appliance, a Virtual Service is defined by a Virtual IP/Port and the protocol. External client requests will be terminated on it, and the APV appliance will load balance the requests to different Real Services.

In Config mode, navigate to Virtual Services.

1. Enter “**vs_bb**” for the Virtual Service Name. Use the check box to enable the virtual service. Select the virtual service type HTTPS from the selector. Set the virtual

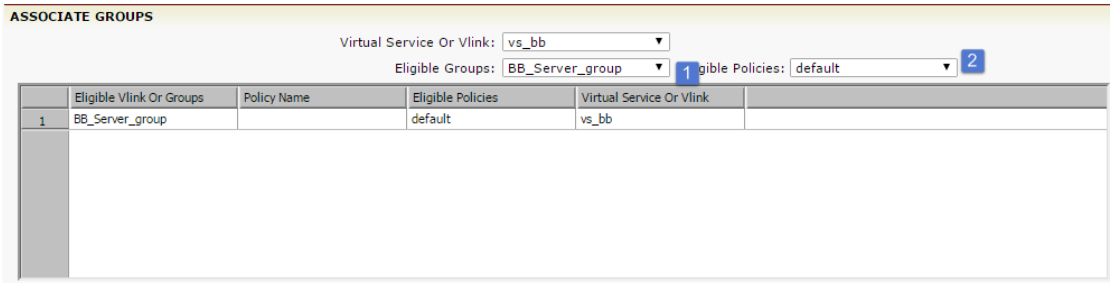
service IP “**10.1.1.73**” and port **443**. Use the check box to enable **ARP**. Set the maximum number of open connections per virtual service. “**0**” means no limitation. Then click “**Add**” to add the APV SLB Virtual Service.

Depending on which type of virtual service is specified, certain parameter fields will appear, change or disappear. Click on the desired action link to add the virtual service. Once a virtual service has been added, it will be displayed within the table. Select a virtual service in the table and double click on it or click on the action link “**Edit**” A new configuration window will present a new series of tabs for completing the virtual service configuration. Select **Add** to save the virtual service.

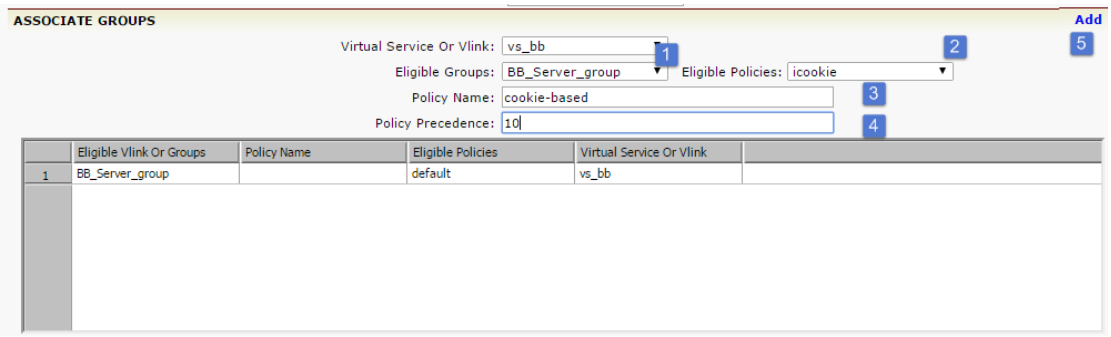


Double click the group in the group list to set up cookie-based load balancing.

2. Select the pre-created “BB_Server_group” from the group list and set “**default**” as the Eligible Policies. Click the “**Add**” button to save this Virtual Service-SLB Group association.

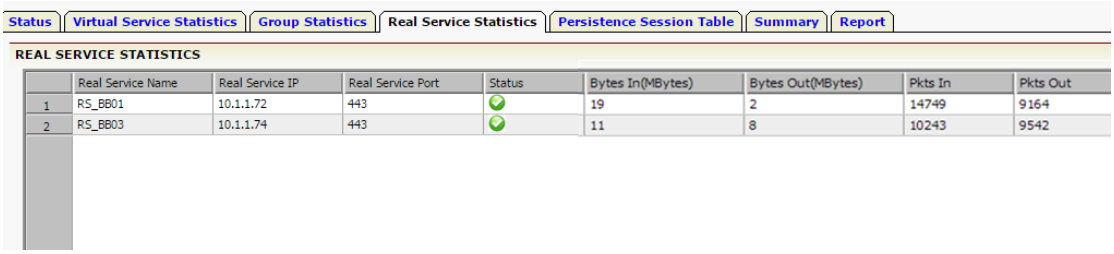


Select “**BB_Server_group**” again to set up a cookie policy as below.



Note: APV SLB supports various Virtual Service Settings. See the Array Support site for documentation if you would like use them for virtual services.

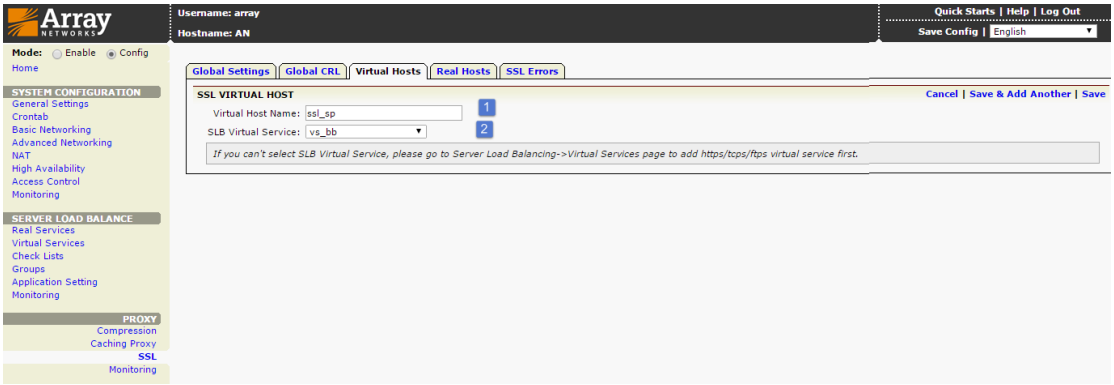
You also can monitor the real service statistics from the APV Web interface:



5.1.5 Create SSL Virtual Hosts

In Config mode, Navigate to **SSL** -> **Virtual Hosts**, and click “**Add**”.

Input the Virtual Host Name (“**ssl_bb**” in the following example) and select the SLB Virtual Service “**vs_bb**”. Then click “**Save**” to store the information.

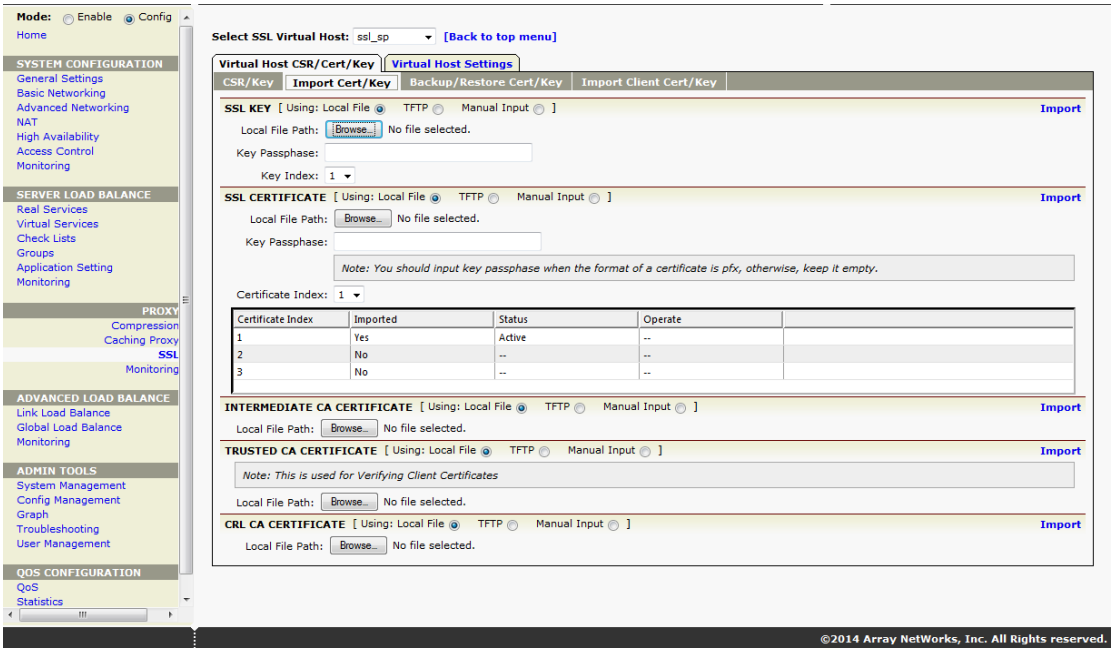


Note: To assign an SSL Certificate/Private Key there are two options:

1. Import an SSL Certificate/Private Key from the backend server (external).
2. Generate a self-signed Certificate (CSR) and Private Key. Send the CSR/Certificate to a public Certificate Authority to sign off, then import it to the APV.

5.1.6 Import an SSL Certificate and Key

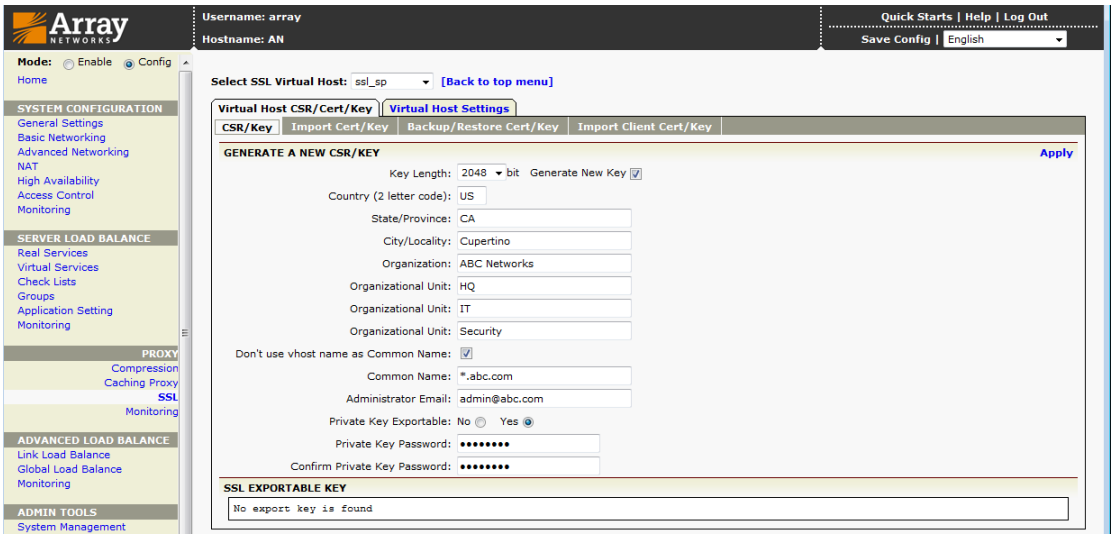
Navigate to **SSL** -> **Virtual Hosts** -> and double click the SSL Virtual Host you just created. Under **Virtual Host CSR/Cert/Key** ->**Import Cert/Key** tab, import the Cert/Key either from a file or via manual import.



5.1.7 Generate a Certificate Signing Request (CSR) and Self-signed Certificate from the APV/vAPV

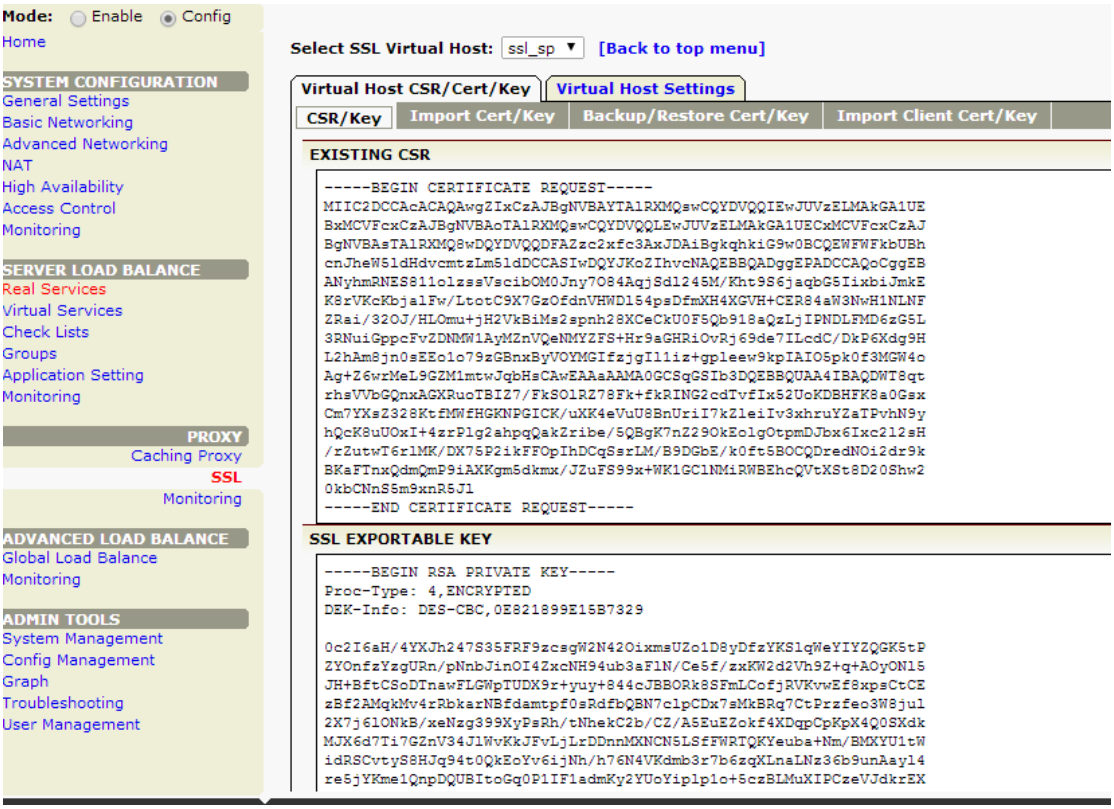
Navigate to **SSL** -> **Virtual Hosts** -> and double click the SSL Virtual Host you just created.

Go to **Virtual Host CSR/Cert/Key** -> **CSR/Key** to generate a CSR and private key. Fill in the proper information and click "**Apply**".

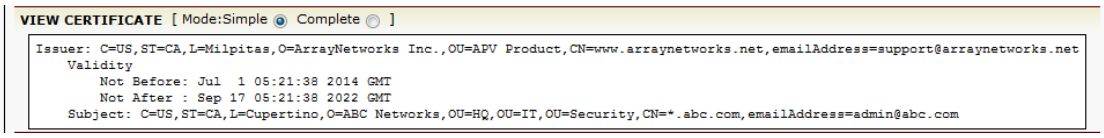


After you have clicked “**Apply**”, the following CSR information will be generated by APV. You can cut and paste the CSR information and email it to a Certificate Authority to have it sign off the certificate.

Before you receive the official SSL certificate, a self-signed SSL certificate is automatically installed and can be used for testing.

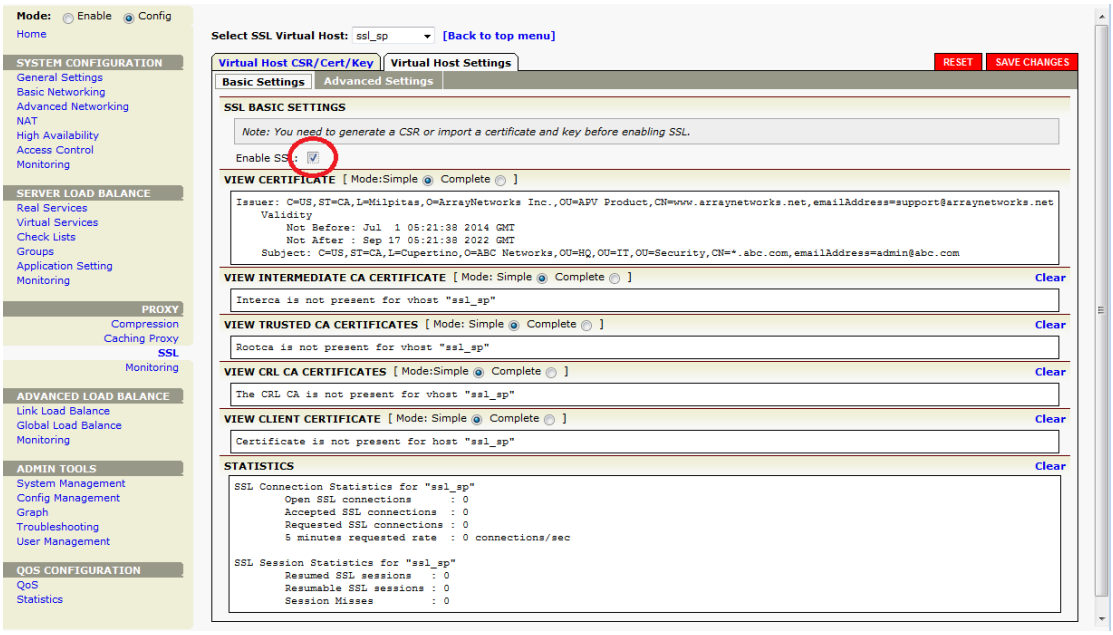


The example below shows a self-signed SSL Certificate.

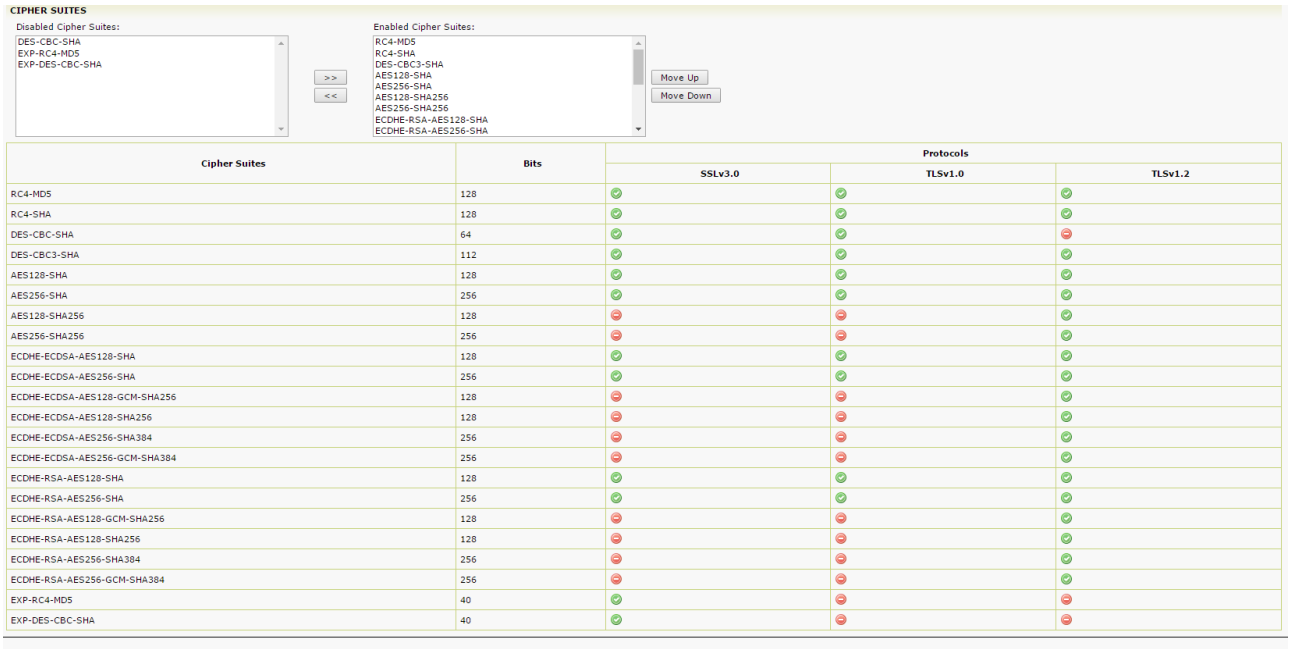


5.1.8 Start SSL

To test (with a self-signed certificate) or run with a production certificate, you will need Enable SSL. Go **SSL->Virtual Hosts** and double click the virtual host “**ssl_bb**”. Select the “**Virtual Host Settings**” tab and select **Enable SSL**.



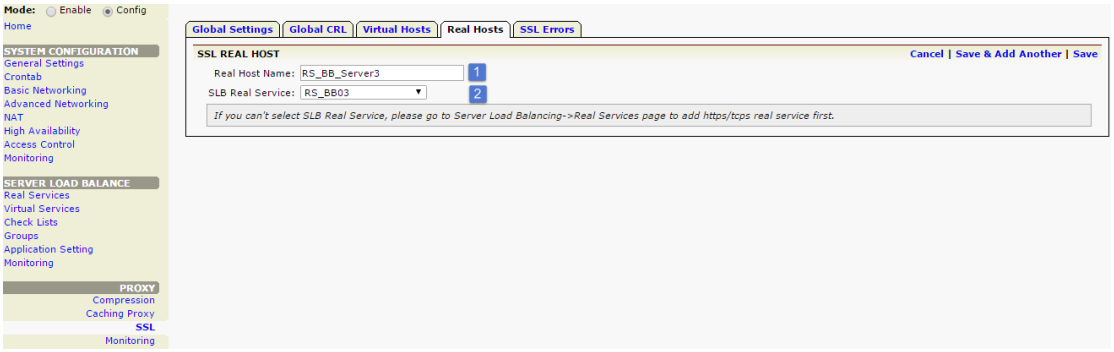
The TLS policies and ciphers can be fine-tuned as below from **SSL -> virtual host setting -> advanced setting.**



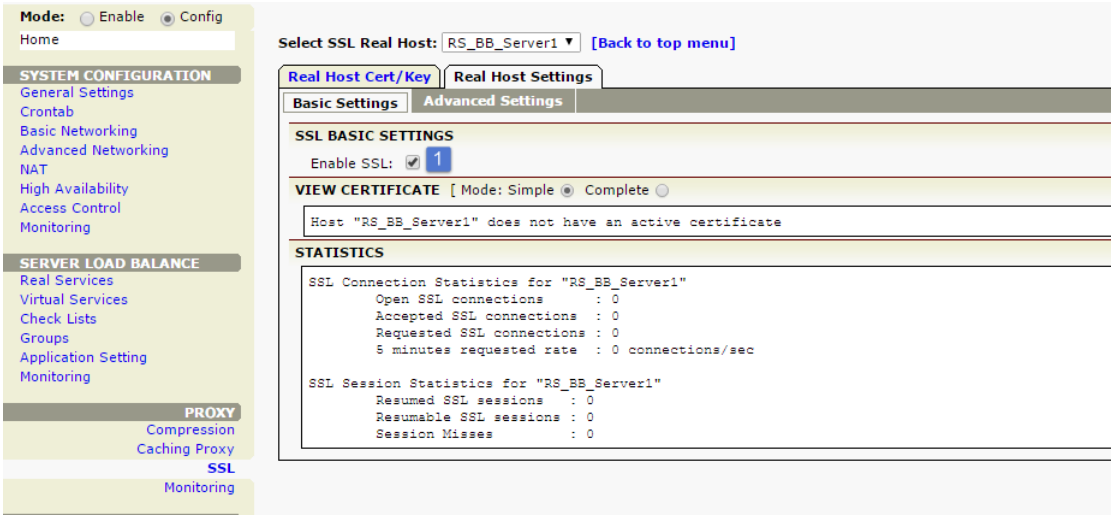
Cipher Suites	Bits	Protocols		
		SSLv3.0	TLSv1.0	TLSv1.2
RC4-MD5	128	✓	✓	✓
RC4-SHA	128	✓	✓	✓
DES-CBC-SHA	64	✓	✓	⊖
DES-CBC3-SHA	112	✓	✓	✓
AES128-SHA	128	✓	✓	✓
AES256-SHA	256	✓	✓	✓
AES128-SHA256	128	⊖	⊖	✓
AES256-SHA256	256	⊖	⊖	✓
ECDHE-ECDSA-AES128-SHA	128	✓	✓	✓
ECDHE-ECDSA-AES256-SHA	256	✓	✓	✓
ECDHE-ECDSA-AES128-GCM-SHA256	128	⊖	⊖	✓
ECDHE-ECDSA-AES128-SHA256	128	⊖	⊖	✓
ECDHE-ECDSA-AES256-SHA384	256	⊖	⊖	✓
ECDHE-ECDSA-AES256-GCM-SHA384	256	⊖	⊖	✓
ECDHE-RSA-AES128-SHA	128	✓	✓	✓
ECDHE-RSA-AES256-SHA	256	✓	✓	✓
ECDHE-RSA-AES128-GCM-SHA256	128	⊖	⊖	✓
ECDHE-RSA-AES128-SHA256	128	⊖	⊖	✓
ECDHE-RSA-AES256-SHA384	256	⊖	⊖	✓
ECDHE-RSA-AES256-GCM-SHA384	256	⊖	⊖	✓
EXP-RC4-MD5	40	✓	⊖	⊖
EXP-DES-CBC-SHA	40	✓	⊖	⊖

5.1.9 Enable Backend/Real Host SSL Service

Add the HTTPS servers to perform the real host mapping for back-to-back SSL.



Enable back-to-back SSL for the Real Host as in the screen shot below.



Input the appropriate “HTTPS” URL to access your Blackboard application server, and make sure you can access every resource from Blackboard.

6 Optional Configuration

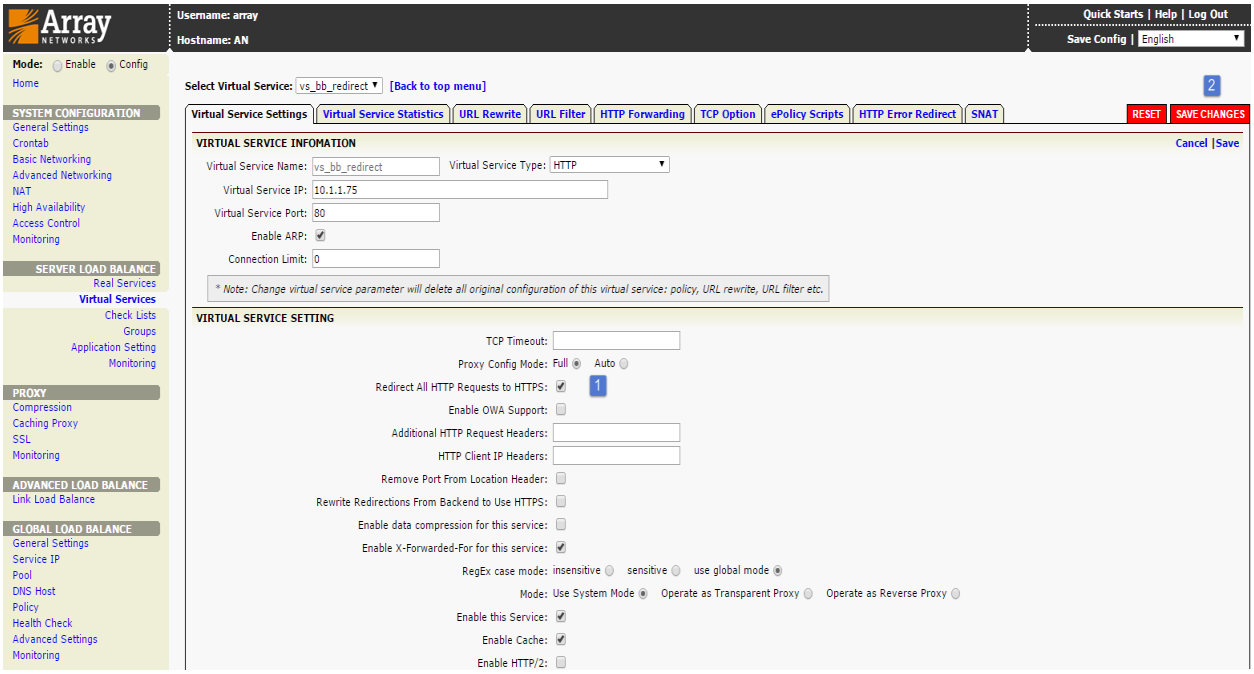
6.1 HTTP Rewrite/Redirect

Users may accidentally type “http://...” (unsecured) instead of “https://...” to access the secured Blackboard server .To make this more user friendly, the APV appliance can be configured to auto redirect http requests to https.

6.1.1 Create another HTTP Virtual Service

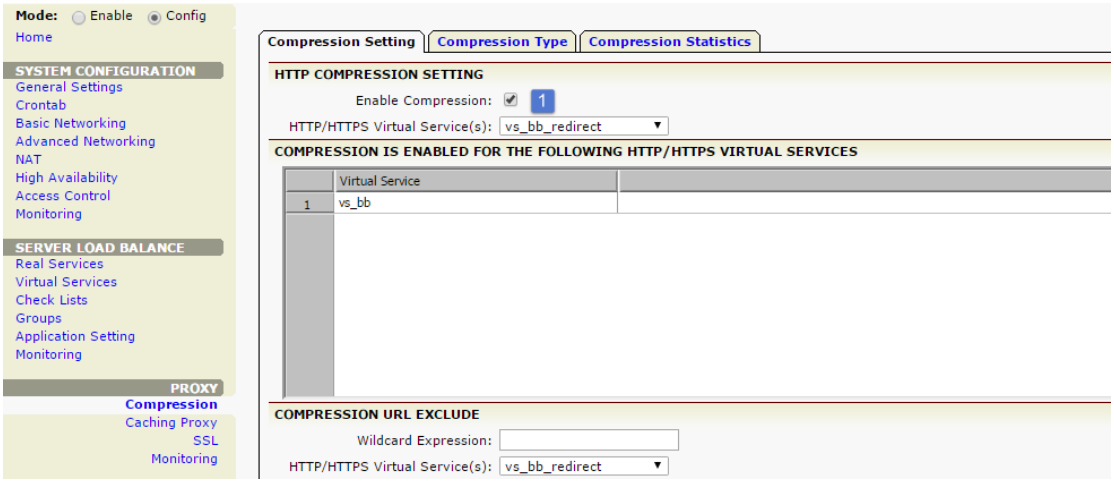
Create another HTTP virtual service and point to the same IP as your HTTPS IP.

Double click the HTTP Virtual Service IP and enable “**Redirect ALL HTTP Requests to HTTPS**”.



6.2 Enable HTTP Compression

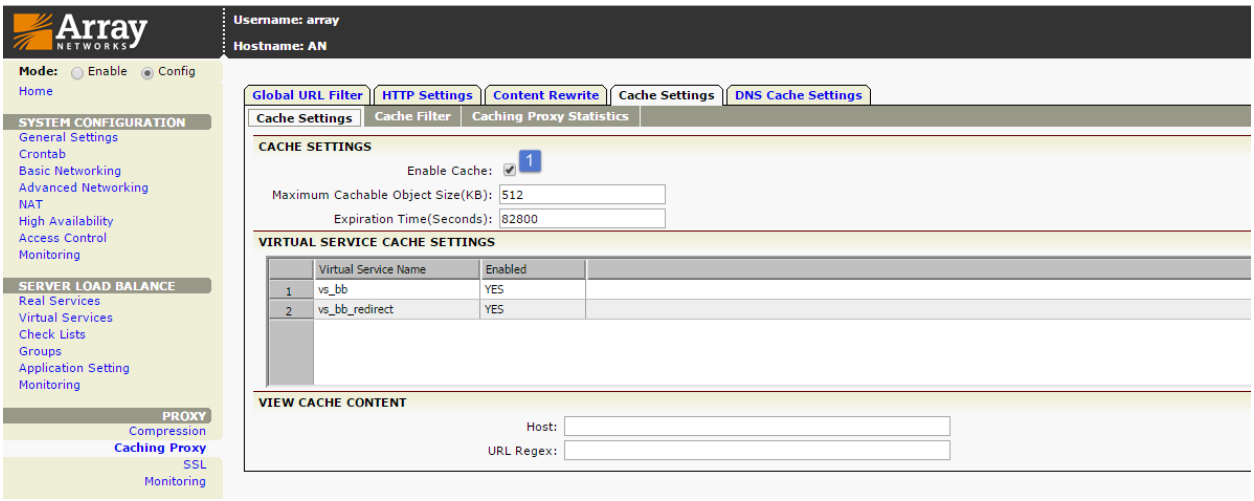
The APV appliance can compress in-line and deliver packet dynamic/static contents over LAN and WAN networks.



Navigate to **Compression** -> **HTTP Compression Setting** to enable the HTTP compression.

6.3 Enable RAM Caching

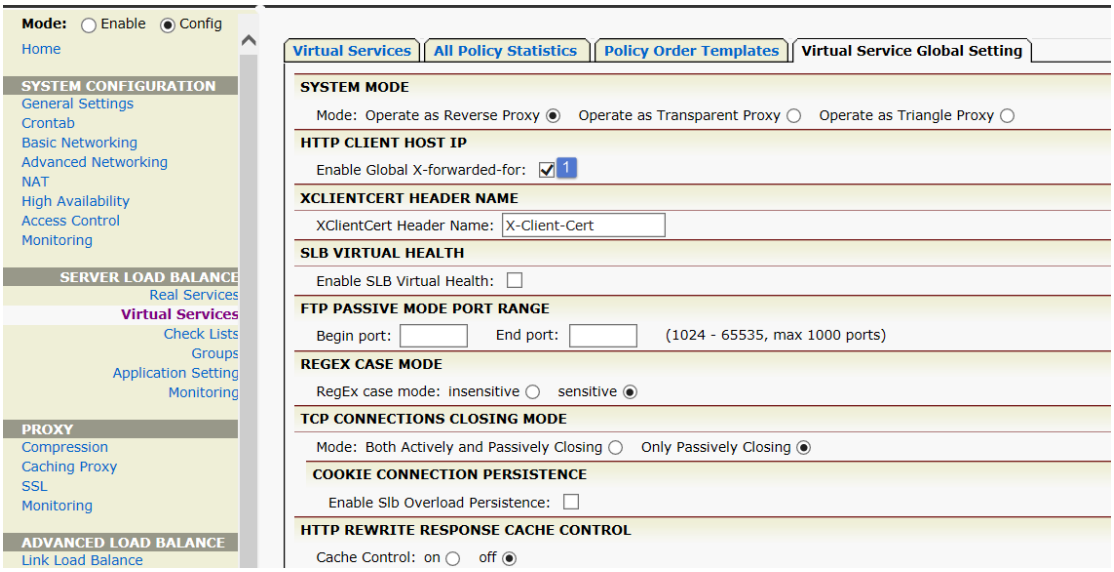
Through RAM caching, the APV appliance serves frequently requested contents from APV memory cache for increased performance and to help scale the capacity of the Blackboard server environment. In addition, a cache rule can be used to utilize client browser cache to further accelerate content delivery and reduce server load.

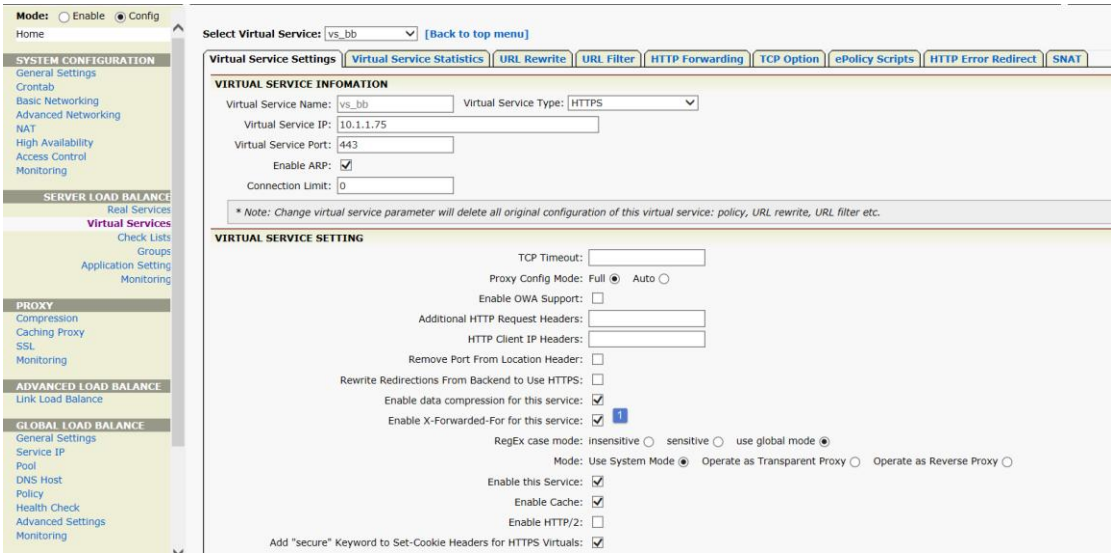


6.4 X-Forwarded-For Header

In a load balanced environment the IP address that is passed is usually the IP address of the load balancer. To preserve the original client IP address most load balancers support the insertion of an X-Forwarded-For header. This should be added when configuring the Load Balancer to ensure the Web application can log the correct user IP

Enable Global x-forwarded-for in Global Settings for the Virtual Services and enable it for the Virtual Service itself.





7 References

http://www.arraynetworks.com/products-application-delivery-controllers-apv-series.html

https://en-us.help.blackboard.com/Learn/9.1_2014_04/Administrator/070_Server_Management_and_Integrations/Performance_Optimization/Load_Balancing_-_Configuration_and_Best_Practices

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
nsedrati@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

May 2016 rev. a