Avaya Port Matrix

Avaya Aura®
Web Gateway

Issue 2
November 18, 2022

ALL INFORMATION IS BELIEVED TO BE CORRECT AT THE TIME OF PUBLICATION AND IS PROVIDED "AS IS". AVAYA INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND FURTHERMORE, AVAYA INC. MAKES NO REPRESENTATIONS OR WARRANTIES THAT THE INFORMATION PROVIDED HEREIN WILL ELIMINATE SECURITY THREATS TO CUSTOMERS' SYSTEMS. AVAYA INC., ITS RELATED COMPANIES, DIRECTORS, EMPLOYEES, REPRESENTATIVES, SUPPLIERS OR AGENTS MAY NOT, UNDER ANY CIRCUMSTANCES BE HELD LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, PUNITIVE, EXEMPLARY, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THE INFORMATION PROVIDED HEREIN. THIS INCLUDES, BUT IS NOT LIMITED TO, THE LOSS OF DATA OR LOSS OF PROFIT, EVEN IF AVAYA WAS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. YOUR USE OF THIS INFORMATION CONSTITUTES ACCEPTANCE OF THESE TERMS.

1. Web Gateway Components

Data flows and their sockets are owned and directed by an application. An Avaya Aura Web Gateway server running on RHEL 8.4 operating system has many components, such as Tomcat Application Server, Telscale SIP servlets, Cassandra, Nginx. For all applications, sockets are created on the server's single network interface. For the purposes of firewall configuration, these sockets are sourced from the server, so the firewall (nftables service) should be running on the same server. Application components in Avaya Aura Web Gateway are listed as follows.

Component	Interface	Description
Nginx	eth0	High-performance HTTP and reverse proxy server. Handles HTTPS connections from Flare clients
Tomcat 8.5.x	lo	Backend HTTP/HTTPS server that serves RESTful requests
Telscale	eth0	Provides a push notification service using a JSR-289 compliant SIP Servlet
Cassandra	eth0	Highly scalable and highly reliable noSQL database
AAWG Administrative Web Interface	eth0	Web interface allowing system administrators to perform server provision, management and licensing-oriented tasks
Serviceability Agent	eth0	The Serviceability Agent is a Java application which receives events and collects inventory information from the product and converts them to its own internal format, encapsulates the message into HTTPS and SNMP traps/informs
Linux keepalived	eth0	Component that provides IP fail-over functionality over VRRP

2. Port Usage Tables

2.1 Port Usage Table Heading Definitions

Source System: System name or type that initiates connection requests.

Source Port: This is the default layer-4 port number of the connection source. Valid values include: 0 – 65535. A "(C)" next to the port number means that the port number is configurable.

Destination System: System name or type that receives connection requests.

Destination Port: This is the default layer-4 port number to which the connection request is sent. Valid values include: 0 – 65535. A "(C)" next to the port number means that the port number is configurable.

Network/Application Protocol: This is the name associated with the layer-4 protocol and layers-5-7 application.

Optionally Enabled / Disabled: This field indicates whether customers can enable or disable a layer-4 port changing its default port setting. Valid values include: Yes or No

"No" means the default port state cannot be changed (e.g. enable or disabled).

"Yes" means the default port state can be changed and that the port can either be enabled or disabled.

Default Port State: The "product" source or destination port is either open, closed, filtered or N/A.

Open: ports will respond to queries

Closed: ports may or may not respond to queries and are listed when they can be optionally enabled.

Filtered: ports can be open or closed, filtered UDP ports will not respond to queries, filtered TCP will respond to queries but will not allow connectivity.

N/A: primarily ephemeral ports used to connect to external sources such as DNS, NTP, etc.

Description: Connection details. Add a reference to refer to the Notes section after each table for specifics on any of the row data, if necessary.

2.2 Port Tables

Below are the tables which document the port usage for this product.

Table 1. Ports for Web Gateway Management Interface (eth0)

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
System administrator's desktop or SAL Gateway	Ephemeral	AAWG	22	TCP/SSH	Yes	Open	System mgmt requiring shell access (See Note 1)
System administrator's desktop or 3rd-party Network Management System (NMS)	Ephemeral	AAWG	161	UDP/SNMP	No	Open	This interface allows a remote SNMP agent to retrieve MIB information
Meetings Management (iView)	Ephemeral	AAWG	3341	TCP/XML	No	Closed	This interface allows a remote Meetings Management (iView) server to establish a non-secure persistent proprietary connection to the Call Signaling Agent component for receiving iView event notifications.
Meetings Management (iView)	Ephemeral	AAWG	3343	TCP/XML	No	Closed	This interface allows a remote Meetings Management (iView) server to establish a non-secure persistent proprietary connection to the Call Signaling Agent component for receiving iView event notifications.
Meetings Management (iView)	Ephemeral	AAWG	3351	TCP/XML	No	Open	This interface allows a remote Meetings Management (iView) server to establish a secure persistent proprietary connection to the Call Signaling Agent component for receiving iView event notifications.
Meetings Management (iView)	Ephemeral	AAWG	3353	TCP/XML	No	Open	This interface allows a remote Meetings Management (iView) server to establish a secure persistent proprietary connection to the Call Signaling Agent component for receiving iView event notifications.
AAWG Clustered Nodes	Ephemeral	AAWG	7000	TCP	Yes	Open	This interface is used by the Cassandra database to replicate data between clustered nodes
AAWG Clustered Nodes	Ephemeral	AAWG	7001	TCP over TLS	Yes	Closed	This interface is used by the Cassandra database to replicate data (over a secure channel) between clustered nodes
Web Browser / Web Management Interface	Ephemeral	AAWG	8445	TCP/HTTPS	No	Open	Avaya Aura Web Gateway Administrative Graphical User Interface. This web-based interface can be used to manage the Avaya Aura Web Gateway application
Meetings Management (iView)	Ephemeral	AAWG	8446	TCP/HTTPS	No	Open	Avaya Aura Web Gateway Administrative Interface. This interface can be used to manage the Avaya Aura Web Gateway application
ALB (Application Load Balancer) in AWS	Ephemeral	AAWG	8457	TCP/HTTP	Yes	Closed	AAWG use ALB for load balancing when running in AWS. The ALB is configured to check whether a node is up by performing an HTTP GET to /health at port 8457. A servlet in Tomcat is used to respond to this. If the servlet is able to respond, that means that both nginx and Tomcat are up.
AAWG Clustered Nodes	Ephemeral	AAWG	8447	TCP/HTTPS	No	Open	This interface is used by the AAWG components for Internode notifications.
AAWG Clustered Nodes	Ephemeral	AAWG	8451	TCP/HTTPS	No	Open	This interface is used by the Telscale component for clustering capability
AAWG Clustered Nodes	Ephemeral	AAWG	9042	TCP/Binary	No	Open	This interface is used by the Cassandra database (client-server communication)
AAWG Clustered Nodes	Ephemeral	AAWG	9160	TCP/Thrift	No	Closed	This interface is used by the Cassandra database (client-server communication)
AAWG	Ephemeral	DNS server	53	UDP/DNS	No	N/A	This interface is used by the Avaya Aura Web Gateway application to perform DNS resolution
AAWG	Ephemeral	DNS server	53	TCP/DNS	No	N/A	This interface is used by the Avaya Aura Web Gateway application to perform DNS resolution
AAWG	Ephemeral	NTP server	123	UDP/NTP	No	N/A	This interface is used by the Avaya Aura Web Gateway application to perform time synchronization (NTP)
AAWG	Ephemeral	Avaya SAL Gateway and/or NMS	162	UDP/SNMP	No	N/A	This interface is used by the Avaya Aura Web Gateway to send SNMP traps to remote Network Management Systems or Avaya Secure Access Link (SAL) Gateway
AAWG	Ephemeral	Avaya Aura System Manager	443	TCP/HTTPS	No	N/A	This interface is used by the Avaya Aura Web Gateway to perform Certificate Management (SCEP) and WebLM licensing
AAWG	Ephemeral	Avaya Aura System Manager	10443	TCP/HTTPS	No	N/A	This interface is used by the Avaya Aura Web Gateway to perform logs forwarding
AAWG	Ephemeral	Customer's LDAP server	389	TCP/LDAP	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to perform user provisioning, authentication and authorization
AAWG	Ephemeral	Customer's LDAP server	636	TCP/LDAP over TLS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to perform user provisioning, authentication and authorization
AAWG	Ephemeral	Customer's AD server	3268	TCP/LDAP	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to perform user provisioning, authentication and authorization
AAWG	Ephemeral	Customer's AD server	3269	TCP/LDAP over TLS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to perform user provisioning, authentication and authorization

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
AAWG	Ephemeral	AAWG Clustered Nodes	3268	TCP	Yes	Open	This interface is used by the Cassandra database to replicate data between clustered nodes
AAWG	Ephemeral	AAWG Clustered Nodes	(1-65535)	TCP over TLS	Yes	Open	This interface is used by the Cassandra database to replicate data (over a secure channel) between clustered nodes
AAWG	Ephemeral	AAWG Clustered Nodes	3269	TCP/Binary	No	Open	This interface is used by the Cassandra database (client-server communication)
AAWG	Ephemeral	AAWG Clustered Nodes	(1-65535)	TCP/Thrift	No	Open	This interface is used by the Cassandra database (client-server communication)
AAWG	Ephemeral	Avaya Aura System Manager	10162	UDP/SNMP	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to send SNMP traps to Avaya System Manager
AAWG	Ephemeral	Avaya Session Border Controller for Enterprise	(1-65535) Default 443	TCP/HTTPS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to get ASBCE load monitoring
AAWG	Ephemeral	Avaya Session Border Controller for Enterprise	8444	TCP/HTTPS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to get ASBCE TURN access data

Table 2. Ports for Web Gateway Application Signaling Interface (eth0)

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Avaya Aura Web Gateway-enabled clients	Ephemeral	AAWG	443	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Web Gateway-enabled clients to consume the Avaya Aura Web Gateway services
Avaya Aura Web Gateway-enabled clients	Ephemeral	AAWG	8443	TCP/HTTPS, TCP/WSS	No	Open	This interface will be used between the Avaya Aura Web Gateway-enabled clients and AAWG for websocket notifcations in case frontend port was not obtained from the Cassandra. Also it is used by the token generation service.
ASBCE or Reverse Proxy	Ephemeral	AAWG	443 (eth0)	TCP/HTTPS	No	Open	This interface is used by reverse proxy or Avaya SBC to route traffic to a particular AAWG node for internal web clients, when the system is configured with a single FQDN
ASBCE or Reverse Proxy	Ephemeral	AAWG	8444 (eth0)	TCP/HTTPS	No	Open	This interface is used by reverse proxy or Avaya SBC to route traffic to a particular AAWG node for external (outside the enterprise) web or mobile clients.
Avaya Aura Session Manager	Ephemeral	AAWG	5060	TCP/SIP	Yes	Closed	This interface is used by the Avaya Aura Web Gateway application to receive SIP messages
Avaya Aura Session Manager	Ephemeral	AAWG	5061	TCP/SIP over TLS	No	Closed	This interface is used by the Avaya Aura Web Gateway application to receive secure SIP messages.
Meetings Management (iView)	Ephemeral	AAWG	5080	TCP/SIP	No	Closed	This interface allows unencrypted SIP communication to be received in the Conferencing-only deployment
Meetings Management (iView)	Ephemeral	AAWG	5081	TCP/SIP over TLS	No	Open	This interface allows encrypted SIP communication to be received in the Conferencing-only deployment
AAWG	Ephemeral	AAWG Clustered Nodes	8448	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Web Gateway for load balanced traffic
AAWG	Ephemeral	Avaya Aura Session Manager / Meetings Management	5060	TCP/SIP	No	N/A	This interface is used by the Avaya Aura Web Gateway application for SIP communications with Avaya Aura Session Manager, Meetings Management and ASBCE
AAWG	Ephemeral	Avaya Aura Session Manager / Meetings Management	5061	TCP/SIP over TLS	No	N/A	This interface is used by the Avaya Aura Web Gateway application for SIP communications over TLS with Avaya Aura Session Manager, Meetings Management and ASBCE
AAWG	Ephemeral	Avaya Aura Presence Services	5269	TCP/XMPP	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to interop with the Avaya Presence Services over XMPP
AAWG	Ephemeral	Avaya Aura Media Server	7151	HTTPS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to interop with the Avaya Aura Media Server
AAWG	Ephemeral	Avaya Aura Device Services	8440	TCP/HTTP over TLS	No	N/A	This interface is used by the Avaya Aura Web Gateway to interop with the Avaya Aura Device Services
AAWG	Ephemeral	Avaya Aura Device Services	8443	TCP/HTTP over TLS	No	N/A	This interface is used by the Portal Client on AAWG to interop with the Avaya Aura Device Services
AAWG	Ephemeral	Meetings Management (iView)	443	HTTPS	No	N/A	This interface is used by the Portal Client on AAWG to interop with the Meetings Management for branding

NOTES:

2.3 Port Table Changes

Table 3. Port Changes From Avaya Aura Web Gateway 3.5 to 3.5.1

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Interface)				
External load balancer	Ephemeral	AAWG	443 (eth0)	TCP/HTTPS	No	Open	This interface is used by external load balancer for redirecting traffic to particular AAWG node
External load balancer	Ephemeral	AAWG	8444 (eth0)	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Web Gateway for detecting traffic coming from external clients through external load balancer

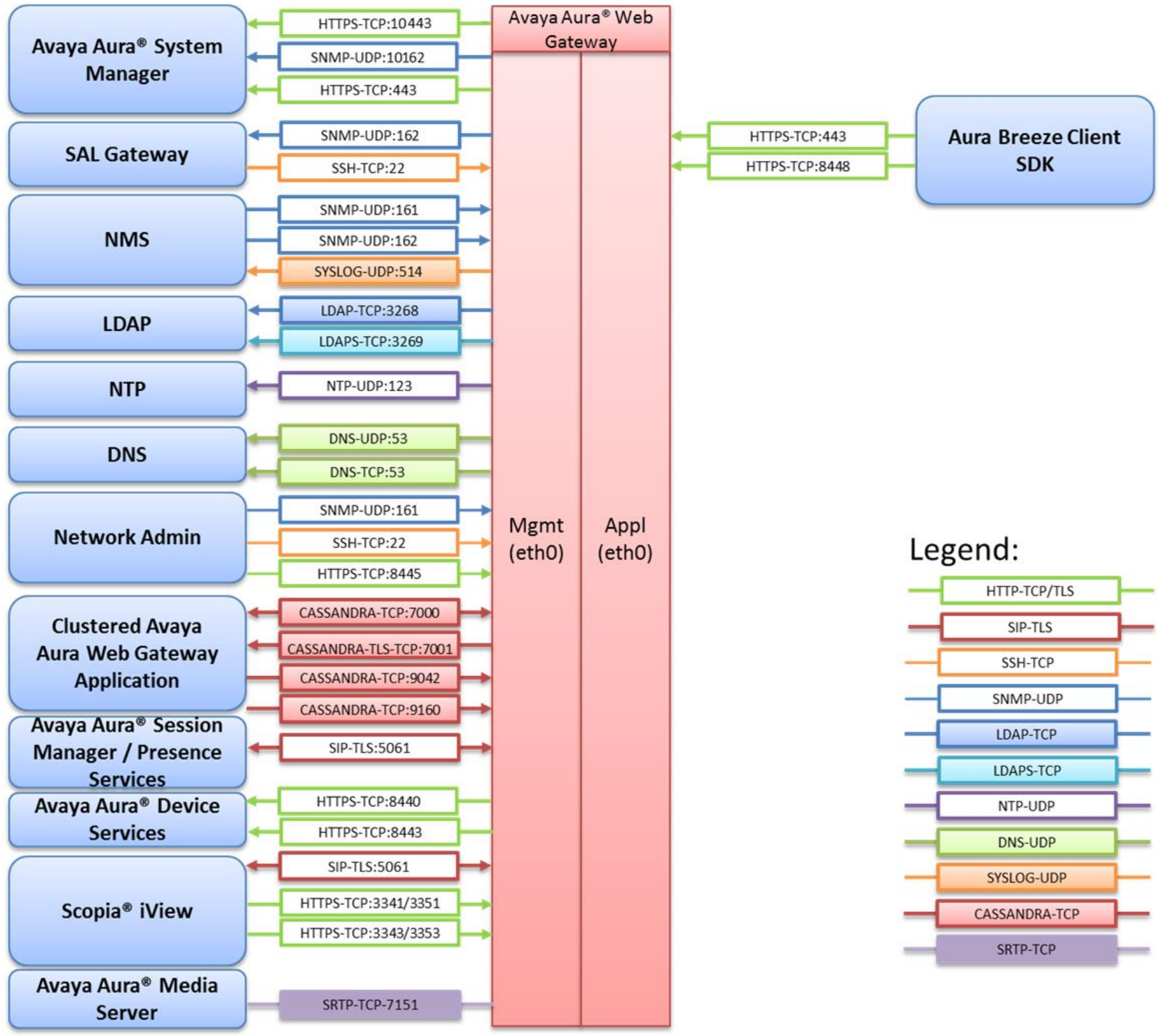
Table 4. Port Changes From Avaya Aura Web Gateway 3.8.1 to 3.9.0

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Interface)				
AAWG	Ephemeral	Avaya Spaces	443 (eth0)	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Web Gateway to interop with the Avaya Spaces

Table 5. Port Changes From Avaya Aura Web Gateway 3.9.0 to 3.11.0

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Interface)				
AAWG	Ephemeral	Gateway Connection Manager	(1-65535) Default 443 (eth0)	TCP/HTTPS TCP/Websocket over TLS	Yes	N/A	This interface is used by the Avaya Aura Web Gateway to interop with the Avaya Cloud Calling

3. Port Usage Diagram



*1 *The image doesn't reflect all ports indicated in the tables*

Appendix A: Overview of TCP/IP Ports

What are ports and how are they used?

TCP and UDP use ports (defined at http://www.iana.org/assignments/port-numbers) to route traffic arriving at a particular IP device to the correct upper layer application. These ports are logical descriptors (numbers) that help devices multiplex and de-multiplex information streams. For example, your PC may have multiple applications simultaneously receiving information: email using destination TCP port 25, a browser using destination TCP port 443 and a ssh session using destination TCP port 22. These logical ports allow the PC to de-multiplex a single incoming serial data packet stream into three mini-streams inside the PC. Each of the mini-streams is directed to the correct high-level application identified by the port numbers. Every IP device has incoming (Ingress) and outgoing (Egress) data streams.

Ports are used in TCP and UDP to name the ends of logical connections which carry data flows. TCP and UDP streams have an IP address and port number for both source and destination IP devices. The pairing of an IP address and a port number is called a socket. Therefore, each data stream is uniquely identified with two sockets. Source and destination sockets must be known by the source before a data stream can be sent to the destination. Some destination ports are "open" to receive data streams and are called "listening" ports. Listening ports actively wait for a source (client) to make contact with the known protocol associated with the port number. HTTPS, as an example, is assigned port number 443. When a destination IP device is contacted by a source device using port 443, the destination uses the HTTPS protocol for that data stream conversation.

Port Types

Port numbers are divided into three ranges: Well Known Ports, Registered Ports, and Dynamic Ports (sometimes called Private Ports). The Well Known and Registered ports are assigned by IANA (Internet Assigned Numbers Authority) and are found here: http://www.iana.org/assignments/port-numbers.

Well Known Ports

Well Known Ports are those numbered from 0 through 1023.
For the purpose of providing services to unknown clients, a service listen port is defined. This port is used by the server process as its listen port. Common services often use listen ports in the well-known port range. A well-known port is normally active meaning that it is "listening" for any traffic destined for a specific application. For example, well known port 23 on a server is actively waiting for a data source to contact the server IP address using this port number to establish a Telnet session. Well known port 25 is waiting for an email session, etc. These ports are tied to a well understood application and range from 0 to 1023.

In UNIX and Linux operating systems, only root may open or close a well-known port. Well Known Ports are also commonly referred to as "privileged ports".

Registered Ports

Registered Ports are those numbered from 1024 through 49151.
Unlike well-known ports, these ports are not restricted to the root user. Less common services register ports in this range. Avaya uses ports in this range for call control. Some, but not all, ports used by Avaya in this range include: 1719/1720 for H.323, 5060/5061 for SIP, 2944 for H.248 and others. The registered port range is 1024 – 49151. Even though a port is registered with an application name, industry often uses these ports for different applications. Conflicts can occur in an enterprise when a port with one meaning is used by two servers with different meanings.

Dynamic Ports

Dynamic Ports are those numbered from 49152 through 65535.
Dynamic ports, sometimes called "private ports", are available to use for any general purpose. This means there are no meanings associated with these ports (similar to RFC 1918 IP Address Usage). These are the safest ports to use because no application types are linked to these ports. The dynamic port range is 49152 – 65535.

Sockets

A socket is the pairing of an IP address with a port number. An example would be 192.168.5.17:3009, where 3009 is the socket number associated with the IP address. A data flow, or conversation, requires two sockets – one at the source device and one at the destination device. The data flow then has two sockets with a total of four logical elements. Each data flow must be unique. If one of the four elements is unique, the data flow is unique. The following three data flows are uniquely identified by socket number and/or IP address.

Data Flow 1: 172.16.16.14:1234 - 10.1.2.3:2345
two different port numbers and IP addresses and is a valid and typical socket pair

Data Flow 2: 172.16.16.14.123**5** - 10.1.2.3:2345
same IP addresses and port numbers on the second IP address as data flow 1, but since the port number on the first socket differs, the data flow is unique

Data Flow 3: 172.16.16.14:1234 - 10.1.2.4:2345

If one IP address octet changes, or one port number changes, the data flow is unique.

Socket Example Diagram

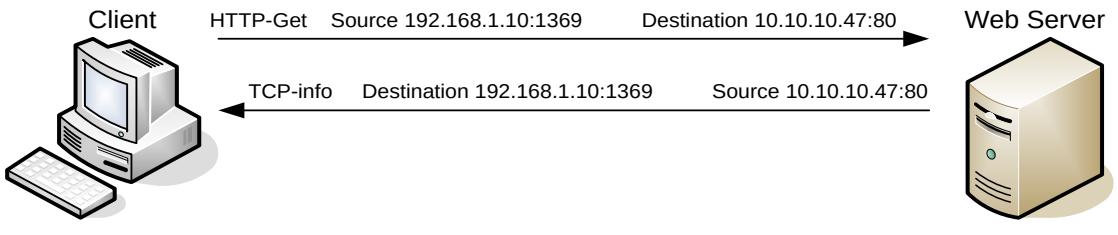


Figure 1. Socket example showing ingress and egress data flows from a PC to a web server

The client egress stream includes the client's source IP and socket (1369) and the destination IP and socket (80). The ingress stream from the server has the source and destination information reversed.

Understanding Firewall Types and Policy Creation

Firewall Types

There are three basic firewall types:

- Packet Filtering
- Application Level Gateways (Proxy Servers)
- Hybrid (Stateful Inspection)

Packet Filtering is the most basic form of the firewalls. Each packet that arrives or leaves the network has its header fields examined against criterion to either drop the packet or let it through. Routers configured with Access Control Lists (ACL) use packet filtering. An example of packet filtering is preventing any source device on the Engineering subnet to telnet into any device in the Accounting subnet.

Application level gateways (ALG) act as a proxy, preventing a direct connection between the foreign device and the internal destination device. ALGs filter each individual packet rather than blindly copying bytes. ALGs can also send alerts via email, alarms or other methods and keep log files to track significant events.

Hybrid firewalls are dynamic systems, tracking each connection traversing all interfaces of the firewall and making sure they are valid. In addition to looking at headers, the content of the packet, up through the application layer, is examined. A stateful inspection firewall also monitors the state of the connection and compiles the information in a state table. Stateful inspection firewalls close off ports until the connection to the specific port is requested. This is an enhancement to security against port scanning[1].

Firewall Policies

The goals of firewall policies are to monitor, authorize and log data flows and events. They also restrict access using IP addresses, port numbers and application types and sub-types.

This paper is focused with identifying the port numbers used by Avaya products so effective firewall policies can be created without disrupting business communications or opening unnecessary access into the network.

Knowing that the source column in the following matrices is the socket initiator is key in building some types of firewall policies. Some firewalls can be configured to automatically create a return path through the firewall if the initiating source is allowed through. This option removes the need to enter two firewall rules, one for each stream direction, but can also raise security concerns.

Another feature of some firewalls is to create an umbrella policy that allows access for many independent data flows using a common higher layer attribute. Finally, many firewall policies can be avoided by placing endpoints and the servers that serve those endpoints in the same firewall zone.

[1] The act of systematically scanning a computer's ports. Since a port is a place where information goes into and out of a computer, port scanning identifies open doors to a computer. Port scanning has legitimate uses in managing networks, but port scanning also can be malicious in nature if someone is looking for a weakened access point to break into your computer.