Avaya Port Matrix

Avaya Aura®
Device Services 10.1.1.0

Issue 1.0
October 03, 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. Avaya Aura Device Services Components

Data flows and their sockets are owned and directed by an application. Here a server running on RHEL 6.6 has many applications, such as JBoss, PostgreSQL, SIP A/S, ASM, etc. For all applications, sockets are created on the network interfaces on the server. For the purposes of firewall configuration, these sockets are sourced from the server, so the firewall (iptables service) should be running on the same server. Application components in the Device Services Application Server are listed as follows.

Component	Interface	Description
Nginx	eth0 (public IP)	Nginx is a high-performance HTTP and reverse proxy server. Used principally here as a web proxy and handles HTTPS connections from clients or other servers.
Tomcat	Lo (localhost)	Tomcat serves as the backend HTTP/HTTPS server that serves RESTful requests, Websockets, and other HTTP-based interfaces. Handles notably: • Contact/Confiuration/Directory requests for AADS clients • administrative Web interface • connecting to the Avaya Aura System Manager for its Data Replication Service
PostgreSQL	eth0 (public IP)	PostgreSQL is a SQL-based database used for storing: • information from the Avaya Aura System Manager Data Replication Service (DRS)
Linux keepalived	eth0 (public IP)	Keepalived is an OS daemon that provides virtual IP and fail-over functionality over VRRP.
SAL-Agent	(Eth0public IP)	The SAL 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 sends it to an Enterprise Server, usually at Avaya.
Apache	Lo (localhost)	Apache acts as file server, for • Setting files (upgrade,46xx) for hard end points • Certificate Files • Firmware Files
Open LDAP	eth0 (public IP)	Open LDAP serves as an Optional LDAP.
Cassandra	eth0 (public IP)	Cassandra is a highly scalable and highly reliable noSQL database, used for storing message metadata and other configuration information.

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: A port is either open, closed or filtered.

Open ports will respond to queries

Closed ports may or may not respond to queries and are only 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.

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 AADS Management Interface (eth0)

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Admin terminal or SAL Gateway	Ephemeral	Avaya Aura Device Services	22	TCP/SSH	Yes	Open	This interface provides Linux shell access for system installation/administration)
Admin terminal or NMS	Ephemeral	Avaya Aura Device Services	161	UDP/SNMP	No	Closed	This interface allows a remote SNMP agent to retrieve MIB information
Avaya System Manager	Ephemeral	Avaya Aura Device Services	2009	TCP/JMX	No	Open	This interface is used to interoperate with the Avaya System Manager Data Replication Service (DRS)
Avaya Aura Device Services	Ephemeral	Avaya System Manager	2009	TCP/JMX	No	Open	This interface is used to interoperate with the Avaya System Manager Data Replication Service (DRS)
Admin terminal (browser)	Ephemeral	Avaya Aura Device Services	8445	TCP/HTTPS	No	Open	Avaya Aura Device Services Administrative Graphical User Interface. This web-based interface can be used to manage the Avaya Aura Device Services application
Avaya Aura Device Services	Ephemeral	DNS server	53	UDP and TCP/DNS	No	Open	This interface is used by the Avaya Aura Device Services application to perform DNS resolution
Avaya Aura Device Services	Ephemeral	Kerberos Server (Domain Controller)	88	TCP/UDP Kerberos	Yes	Open	This interface is used by the Avaya Aura Device Services application to perform Kerberos authentication for Integrated Windows Authentication (IWA)
Avaya Aura Device Services	Ephemeral	NTP time server	123	UDP/NTP	No	Open	This interface is used by Avaya Aura Device Services application to perform time synchronization (NTP)
Avaya Aura Device Services	Ephemeral	Avaya SAL Gateway and/or NMS	162	UDP/SNMP	No	Open	This interface is used by the Avaya Aura Device Services to send SNMP traps to remote Network Management Systems or Avaya Secure Access Link (SAL) Gateway

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Avaya Aura Device Services	Ephemeral	Avaya SAL Gateway and/or NMS	514	UDP/SNMP	No	Open	This interface is used by the Avaya Aura Device Services for SYSLOG interface with remote Network Management Systems or Avaya Secure Access Link (SAL) Gateway
Avaya Aura Device Services	Ephemeral	Avaya Aura System Manager	443	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Device Services for SYSLOG interface with remote Network Management Systems or Avaya Secure Access Link (SAL) Gateway
Avaya Aura Device Services	Ephemeral	Avaya Spaces	443	TCP/HTTPS	Yes	Open	This interface is used by the Avaya Aura Device Services to interoperate with Avaya Spaces.
Avaya Aura Device Services	Ephemeral	LDAP Server	3268 configurable	TCP/LDAP	Yes	Closed	This interface is used by the Avaya Aura Device Services to perform user provisioning, authentication and authorization
Avaya Aura Device Services	Ephemeral	LDAP Server	3269 configurable	TCP/LDAP over TLS	Yes	Open	This interface is used by the Avaya Aura Device Services to perform user provisioning, authentication and authorization
Avaya Aura Device Services	Ephemeral	Avaya System Manager	10162	UDP/SNMP	No	Open	This interface is used by the Avaya Aura Device Services to send SNMP traps to Avaya System Manager
Avaya Aura Device Services (remote)	Ephemeral	Avaya Aura Device Services	8441	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Device Services server to interoperate with other Avaya Aura Device Services servers in a multi-site deployment
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8447	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Device Services application to enable Server to Server Communication in a Clustered environment
Device Services Clients	Ephemeral	Avaya Aura Device Services	443/8443	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Device Services -enabled clients to consume the device services
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8448	TCP/HTTPS	No	Open	This interface is used by the Avaya Aura Device Services application for load-balancing purposes
Avaya System Manager	Ephemeral	Avaya Aura Device Services	8448	TCP/HTTPS	No	Open	This interface is used to interoperate with the Avaya System Manager Data Replication Service (DRS)
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8442	TCP/HTTPS	Yes	Closed	This interface is used by Avaya Aura Device Services to pass web deployment requests without certificate validation.

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Avaya Aura Web Gateway (AAWG)	Ephemeral	Avaya Aura Device Services	8440	TCP/HTTPS	No	Closed	This interface is used by Avaya Aura Device Services application for accepting the server-to-serve request via Optional certificate policy for AAWG connections.
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	1543	TCP/HTTPS	Yes	Open	This interface is used by Avaya Aura Device Services to serve the 46xxsettings.txt, *Upgrade*.txt files, Certificate Files, Firmware files etc. required by hard endpoints
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8543	TCP/HTTPS	Yes	Open	This interface is used by Avaya Aura Device Services to serve the admin interface of Utility Server
Equinox Management	Ephemeral	Avaya Aura Device Services	3354	TCP/HTTPS	No	Open	This interface is used by the Equinox Management server to establish a persistent connection to Avaya Aura Device Services to enable event notification and synchronization of conference user information.
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	80	TCP/HTTP	Yes	Closed	This interface is used by Avaya Aura Device Services to serve the Firmware updates for H323 Devices
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8080	TCP/HTTP	Yes	Open	This interface is used by Avaya Aura Device Services to serve the 46xxsettings.txt, *Upgrade*.txt files, Certificate Files, Firmware files etc. required by hard endpoints. Note that this port is enabled for local access.
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8082	TCP/HTTP	Yes	Open	This interface is used by Avaya Aura Device Services to serve the Firmware updates for H323 Devices
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8468	TCP/HTTP	Yes	Open	This interface is used by Avaya Aura Device Services as the non-load balance port for the Firmware updates requests from H323 Devices
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8458	TCP/HTTPS	Yes	Open	This interface is used by Avaya Aura Device Services as the non-load balance port to serve the 46xxsettings.txt, *Upgrade*.txt files, Certificate Files, Firmware files, Phone backup Files required by hard endpoints

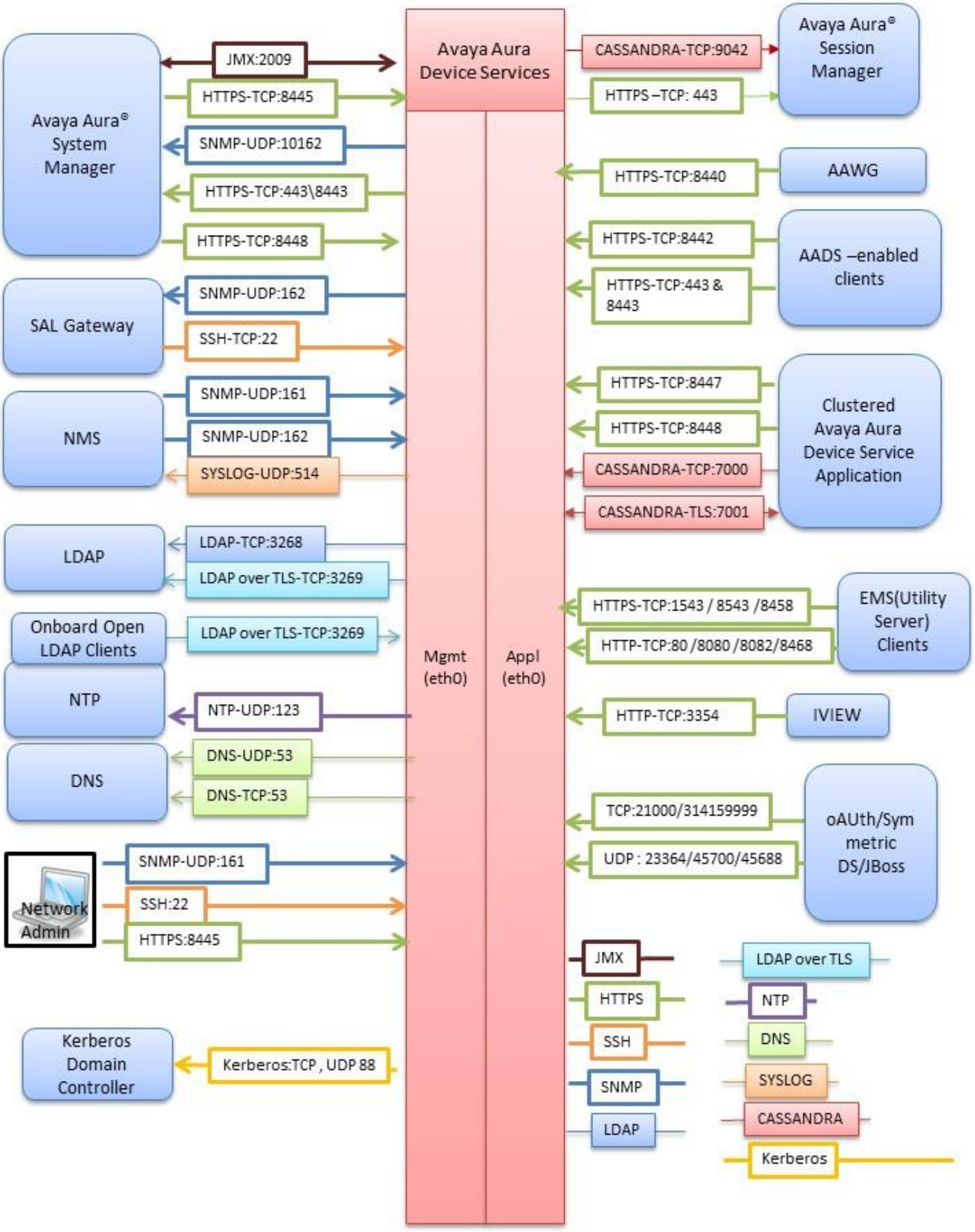
Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	8457	HTTP	No	Open	This interface is used by the Avaya Aura Device Services to serve the health status of the system.
Avaya Aura Device Services Cassandra	Ephemeral	Avaya Aura Device Services	7000	TCP	No	Open	This interface is used by the Cassandra database to replicate data between clustered nodes
Avaya Aura Device Services Cassandra	Ephemeral	Avaya Aura Device Services	7001	TCP/TLS	No	Open	This interface is used by the Cassandra database to replicate data (over a secure channel) between clustered nodes
Avaya Aura Device Services Onboard OpenLDAP	Ephemeral	Avaya Aura Device Services	3269	TCP/TLS	Yes	Open	This interface is used by the Avaya Aura Device Services to serve Onboard OpenLDAP(optional) request from the respective Clients
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	21000	TCP/TLS	Yes	Open	This interface is used by the Avaya Aura Device Services to serve Symmetric DS services for OAuth Feature
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	31415	TCP/TLS	Yes	Open	This interface is used by the Avaya Aura Device Services to serve Symmetric DS services for OAuth Feature
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	23364	UDP	Yes	Open	This interface is used by the Avaya Aura Device Services to serve JGROUP services for OAuth Feature. This port is used by OAUTH for JBOSS mod cluster multicast address
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	45700	UDP	Yes	Open	This interface is used by the Avaya Aura Device Services to serve JGROUP services for OAuth Feature. This UDP port is used by OAUTH for jgroup mping multicast address
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	45688	UDP	Yes	Open	This interface is used by the Avaya Aura Device Services to serve JGROUP services for OAuth Feature. This port is used by OAUTH for jgroup UDP multicast address
Avaya Aura Device Services	Ephemeral	Avaya Aura Device Services	9999	UDP	Yes	Open	This interface is used by the Avaya Aura Device Services to serve Jboss management services for OAuth Feature. This port is used by OAUTH for slave node to talk to master node of Keycloak

Source		Destination		Network / Application Protocol	Optionally Enabled / Disabled?	Default Port State	Description
System	Port (Configurable Range)	System	Port (Configurable Range)				
Avaya Aura Device Services	Ephemeral	Avaya Session Manager	9042	TLS / Proprietary	No	Open	This interface is used by Avaya Aura Device Services to retrieve contact information from the Avaya Session Manager Cassandra database (management interface).
Avaya Aura Device Services	Ephemeral	Avaya Session Manager	443	TLS/XML	No	Open	This interface is used by Avaya Aura Device Services to Create, Update and Delete contacts from a user's PPM contact list on Avaya Session Manager (asset interface)

2.3 Port Table Changes

There are no port table changes

3. Port Usage Diagram


Avaya Aura® System Manager
JMX:2009
HTTPS-TCP:8445
SNMP-UDP:10162
HTTPS-TCP:443\8443
HTTPS-TCP:8448
SAL Gateway
SNMP-UDP:162
SSH-TCP:22
NMS
SNMP-UDP:161
SNMP-UDP:162
SYSLOG-UDP:514
LDAP
LDAP-TCP:3268
LDAP over TLS-TCP:3269
Onboard Open LDAP Clients
LDAP over TLS-TCP:3269
NTP
NTP-UDP:123
DNS
DNS-UDP:53
DNS-TCP:53
Network Admin
SNMP-UDP:161
SSH:22
HTTPS:8445
Kerberos Domain Controller
Kerberos:TCP , UDP 88
Avaya Aura Device Services
Mgmt (eth0)
Appl (eth0)
CASSANDRA-TCP:9042
HTTPS –TCP: 443
Avaya Aura® Session Manager
HTTPS-TCP:8440
AAWG
HTTPS-TCP:8442
HTTPS-TCP:443 & 8443
AADS –enabled clients
HTTPS-TCP:8447
HTTPS-TCP:8448
CASSANDRA-TCP:7000
CASSANDRA-TLS:7001
Clustered Avaya Aura Device Service Application
HTTPS-TCP:1543 / 8543 /8458
HTTP-TCP:80 /8080 /8082/8468
EMS(Utility Server) Clients
HTTP-TCP:3354
IVIEW
TCP:21000/314159999
UDP : 23364/45700/45688
oAUth/Symmetric DS/JBoss
JMX
HTTPS
SSH
SNMP
LDAP
LDAP over TLS
NTP
DNS
SYSLOG
CASSANDRA
Kerberos

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

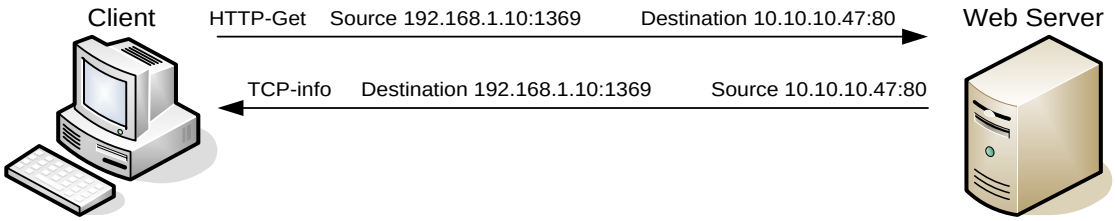


Figure 1. Socket example showing ingress and egress data flows from a PC to a web server

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.