



Operation

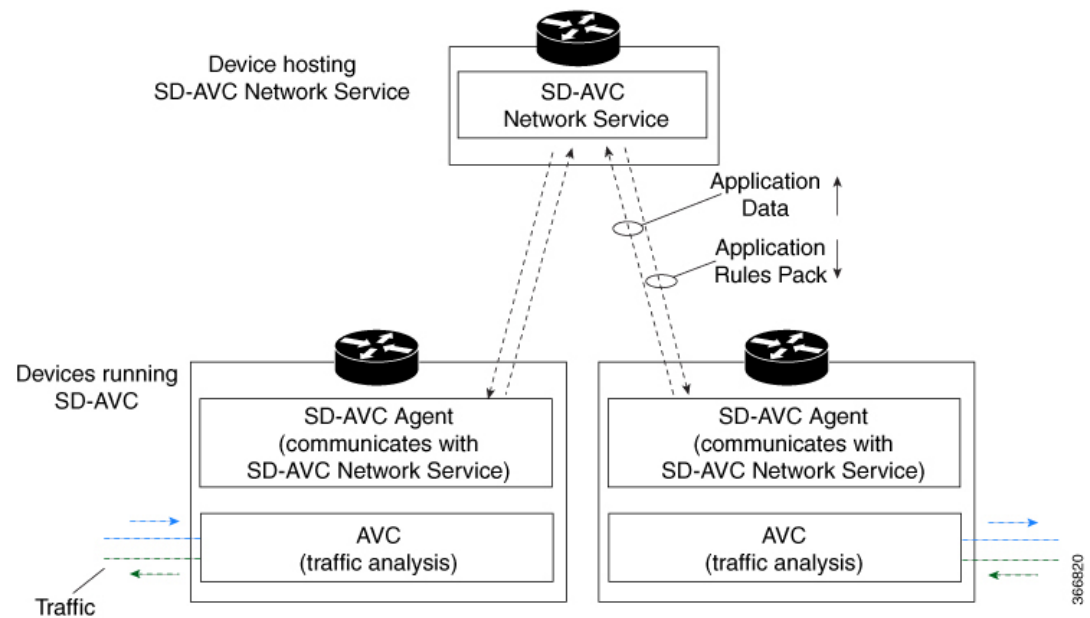
- [SD-AVC Architecture, page 1](#)
- [SD-AVC and Application Recognition, page 2](#)

SD-AVC Architecture

SD-AVC architecture consists of two basic components:

- Centralized SD-AVC network service component operating on a host device
- SD-AVC Agent component running on each SD-AVC-enabled device in the network

Figure 1: SD-AVC Network Service and Agents



SD-AVC and Application Recognition

Cisco AVC can recognize 1400+ network applications, providing recognition of most enterprise network traffic. SD-AVC offers a controller-based approach that operates network-wide, aggregating application information collected across the network, and centralized deployment of Protocol Pack updates.

SD-AVC improves application recognition, and offers a solution to challenges posed by complex networks that use a variety of routing devices and routing methods. Such challenges include asymmetric routing, first packet classification, encryption, and so on.

Collecting Application Data

Devices in the network running AVC analyze traffic and generate application data. If a device is connected to SD-AVC, the SD-AVC agent operating on the device receives this application data, and processes and caches the data. Periodically, the SD-AVC agent sends the latest application data to the centralized SD-AVC network service.

As new servers are detected or as server addresses change, the agent continually discovers and validates these servers and updates the SD-AVC network service with the new information. The process of discovery and validation can take several minutes.

Server addresses usually remain constant over time, but when they do change, the SD-AVC agent detects the changes and updates the network service.

Aggregating Application Data

The SD-AVC network service aggregates application data from multiple sources, producing an application rules pack from the composite data. This is made available to network devices using SD-AVC.

Periodically, the network devices using SD-AVC request the application rules pack. Relying on devices to pull (request) the application rules pack on their own schedule improves efficiency and simplifies administration.

The application rules pack contains the following type of information: ID, IP address, port, network protocol, VRF name, application name, and so on.

Example:

```
ID      | IP Address | Port | Protocol | VRF-name | App-Name
=====
0      | 192.0.2.1 | 5901 | TCP      | Mgt      | VNC
```