Cisco Spark

Firewall Traversal

Version 1.0

Cisco Spark is a cloud collaboration platform that brings together all your people and communication tools in one secure and easy-to-use app. With the Cisco Spark app, you can send messages, share files, and meet with different teams, all in one place. Cisco Spark runs on various devices, such as your smartphones, laptops, Cisco room devices, and Cisco Spark Boards.

In this white paper, we discuss what is needed for a successful deployment of Cisco Spark in a network behind a typical enterprise firewall.

You'll also learn how a typical customer journey with Cisco Spark operates. This journey starts with early adopters who want to test Cisco Spark. It's important that early adopters get Cisco Spark to work without too much hassle. As early adopters go through trials, more official trial programs, and later to full deployment, we need to make sure that their enterprise network configuration is optimized for the best possible user experience with Cisco Spark.

The examples in this white paper refer to real URLs, IP ranges, and port usage. To get the most accurate and current information, see the online Cisco Spark firewall documentation.[1]

---

[1] https://support.ciscospark.com/customer/en/portal/articles/1911657-firewall-and-network-requirements-for-the-cisco-spark-app

Table of Contents

Customer Journey

You don't need to host and maintain a large set of complicated services on-premises anymore. With the Cisco Spark app, you can connect to Cisco Collaboration Cloud to handle all your messaging, video conferencing, interoperability with legacy VoIP systems, and so on. The list of features that Cisco Collaboration Cloud supports is rapidly growing.

Cisco Collaboration Cloud offers meetings-focused services. What that means is that when the Cisco Spark app connects to the cloud, you get features like gapless Wi-Fi/cellular handoffs and the ability to move calls from mobile to video units. This scenario is possible because all calls go through the Cisco Collaboration Cloud.

Cisco Spark communicates with Cisco Collaboration Cloud through HTTPS (including WebSockets) for messaging and signaling, and the Secure Real-Time Transport Protocol (SRTP) for media. To keep latency to a minimum, Cisco Spark uses the User Datagram Protocol (UDP) as the preferred transport protocol for interactive media. With Cisco Collaboration Cloud, new nodes can be close to customer locations, which helps to keep latency low.

Some enterprises might have restrictive security policies that prevent the Cisco Spark app from connecting optimally to Cisco Collaboration Cloud. For that reason, we have a "Get it working now, optimize later" strategy that simplifies our trials process for our early adopters. See Figure *1*.

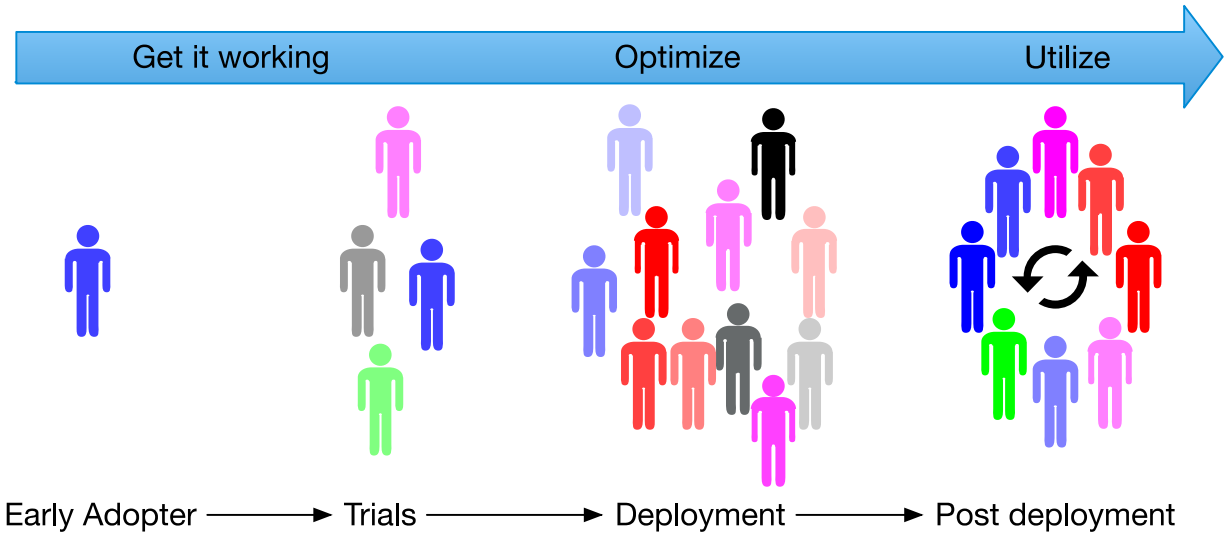

Figure 1 Early Adoption to Deployment

You can learn about a successful Cisco Spark app customer journey, from early adoption to post deployment, in the next chapter.

Early Adoption

Some early adopters want to try Cisco Spark but they find that their IT departments restrict connections to Cisco Collaboration Cloud. How did we remove this obstacle so that early adopters can successfully try out Cisco Spark? That's why we developed our "Get it working now, optimize later" strategy to simplify the trials process for our early adopters.

If early adopters can browse the Internet from their enterprise network, they can use the Cisco Spark app, unless their enterprise actively blacklists any of the needed URLs that the Cisco Spark application needs to connect to.

The Cisco Spark app supports various options that allow early adopters to connect to Cisco Collaboration Cloud. For the best possible media quality, we recommend that early adopters open up their firewall for outbound UDP traffic. If that's not possible, the Cisco Spark app supports TCP/TLS fallback for media. As a last resort, it's also possible that early adopters can send media through HTTP proxies.[2] Note that if you send media through a proxy, it often has a negative impact on audio/video quality, but this option allows for simple trials. Our goal for our early adopters is to help them get something up and running that can be optimized later.

Official Trials

At this stage, the early adopter's IT department is involved. Early adopters can do simple changes to the network infrastructure configuration such as configuring the firewall and making sure that enough bandwidth is available to satisfy the requirements for good quality media.[3]

Another option at this stage is to install a local Cisco Spark Hybrid Media Node.[4] This node is a virtual image that you can install in the local network. The media traffic from the Cisco Spark app can terminate on the hybrid media nodes within your network instead of terminating on a media node in Cisco Collaboration Cloud.

If early adopters open the UDP port on the firewall for media (our recommended solution), they should also make sure that any Intrusion Detection Systems (IDS) installed in the network are aware of this change, because a sudden spike in UDP traffic can trigger alarms.

---

[2] Currently, Cisco Spark Board, and the Cisco DX Series/Cisco SX Series do not support media through HTTP proxies.

[3] https://help.webex.com/docs/DOC-7231_-reference_08E4325F587B50ADEECD05E3CDC9D16C

[4] http://www.cisco.com/go/hybrid-services-media

If early adopters can't allow UDP traffic through their firewall at this stage, we recommend that they don't send their media through an HTTP proxy. HTTP proxies can add latency and might not handle the required bandwidth needed for a successful official trial. If early adopters can't allow UDP through their firewall for media, we recommend that they use TCP/TLS fallback or a local virtual hybrid media node so that the Cisco Spark app can send media over UDP.[5]

Deployment

In this phase, Cisco Spark becomes an integral part of the early adopter's organization. At this stage, the early adopter's IT department actively helps to optimize the network by choosing deployment options that are based on their own network requirements and security policies. For example, the early adopter can benefit from low latency meetings media by opening inside-initiated UDP connections to ANY IP on port 5004. See Figure 2.

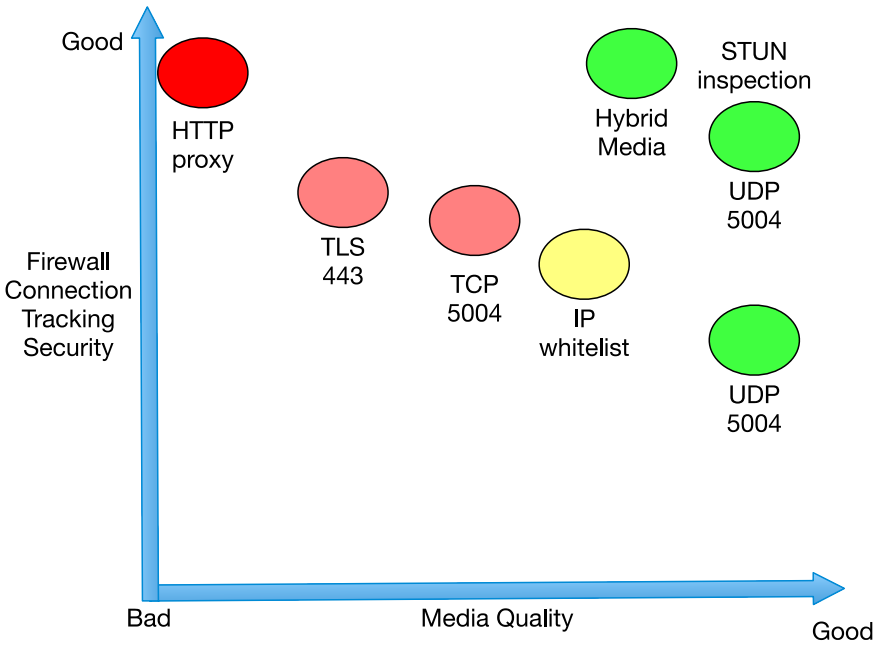

Figure 2 Security Versus Media Quality

Cisco Spark always tries to reach a Cisco Collaboration Cloud media node that can provide the best media quality. If early adopters are concerned about opening inside-initiated UDP connections to ANY on port 5004, we recommend that they look into the Session Traversal Utilities for NAT (STUN) inspection capabilities of the installed firewall. STUN inspection

[5] Cisco Spark Board does not support TCP/TLS fallback for media

adds TCP SYN/ACK-like properties in the initial packets. The Cisco Adaptive Security Appliance firewall supports this feature.[6]

Other options that an early adopter can consider is to include an IP range whitelist and install a local Cisco Spark Hybrid Media Node to handle media.

Transport Layer Security (TLS) does not add any extra protection in securing the data in the media stream. The media stream payload is always encrypted with the Secure Real Time Transport Protocol (SRTP).

Post Deployment

Technology changes and evolves, along with your enterprise requirements and needs. Your journey does not end after deployment; your post-deployment journey with Cisco Spark is just beginning. For more information, contact your Cisco Customer Success representative.

[6] http://www.cisco.com/c/en/us/td/docs/security/asa/asa96/configuration/firewall/asa-96-firewall-config/inspect-voicevideo.html - id_17622

Messages and Signaling

Messaging and meeting setup requires the Cisco Spark app or endpoints to connect to Cisco Collaboration Cloud. This chapter describes how you can configure a firewall to allow the Spark app to connect to Cisco Collaboration Cloud.

Ports and Protocols

Cisco Collaboration Cloud is deployed in data centers across the world. The Cisco Spark app or endpoint can reach the cloud only if it can open a connection out on TLS port 443. The traffic on TLS port 443 is either HTTPS or WebSocket. If your normal web browsing works, the Cisco Spark app will also work.

URLs

Cisco Spark connects to four categories of URLs:

- Messaging and signaling
- File and content storage
- Software upgrade images
- Metric and analytics

Messaging and Signaling

The Cisco Spark app can connect to Cisco owned and controlled URLs for messaging, signaling, and identity management. Typically, the URLs look as follows:[7]

idbroker.webex.com

identity.webex.com
*.wbx2.com
*.ciscospark.com

File and Content Storage

The Cisco Spark app can connect to these URLs for file and content storage:

[7] https://help.webex.com/docs/DOC-7231_-reference_00E0C01DC4F01F91012A027BE9E4288B

*.clouddrive.com

*.rackcdn.com

All Cisco Spark content is encrypted end-to-end and is safe both in transit and at rest. The content decryption keys are located at a Cisco-operated Key Management Server (KMS) or locally at an enterprise KMS. See the Cisco Spark Security white paper for more details.[8]

Software Upgrades

The Cisco Spark app has a frequent update cycle. Security fixes and other continuous improvements are automatically updated by the Cisco Spark app or endpoint. To ensure that you can download our upgrades easily, we rely on a Content Delivery Network (CDN). The Cisco Spark app automatically downloads updates. Some devices require user action to install; other devices install the update automatically.[9]

Metrics and Analytics

Collecting metrics and analytics is key for Cisco Spark. This data collection adheres to the Cisco Privacy Statement.[10]

It's important that we collect this data because it helps us fine-tune our services. With this data, we can optimize data center locations and internal networks to make sure that latency is low.

Cisco Spark uses these URLs to send crash reports and usage metrics:

*.crashlytics.com
*.mixpanel.com

---

[8] http://www.cisco.com/c/dam/en/us/solutions/collateral/collaboration/cloud-collaboration/cisco-spark-security-white-paper.pdf
[9] https://support.ciscospark.com/customer/portal/articles/1335068-update-the-cisco-spark-app-to-the-latest-release
[10] http://www.cisco.com/c/en/us/about/legal/privacy-full.html

Cisco Spark can also connect to other URLs, depending on which Cisco Spark app or endpoint is in use. For more information, see the online Cisco Spark documentation.[11]

[11] https://help.webex.com/docs/DOC-7231

Sending and Receiving Media

To enable audio and video with the best possible quality, we recommend that your firewall allows inside-initiated UDP connections to port 5004. If the Cisco Spark app fails to connect on UDP, it can fall back to traverse a firewall using TCP port 5004 and TLS port 443 as well, but the media quality might suffer.

The Cisco Spark app also supports a media tunnel through an HTTP proxy. See Figure *3*. This tunnel can help you to maintain connectivity in scenarios where an HTTP proxy is required and no other protocols succeed. The drawback is that it usually causes poor media quality.

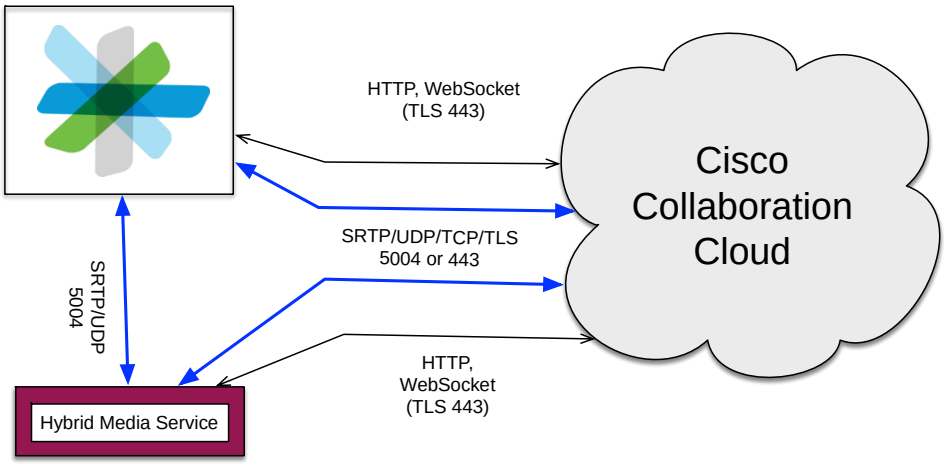

Figure 3 Cisco Collaboration Cloud Ports and Protocols

All connections are initiated by the Cisco Spark app. This requirement is important because it allows the Cisco Spark app to work in Network Address Translation (NAT) environments. The Cisco Spark app or endpoint expects the firewall's NAT table entry to expire after some time of inactivity and sends keepalives to maintain the connection if it's still needed. How long a firewall/NAT actually keeps the pinhole open varies. Media is always symmetric; the same source port is used to send and receive media.

Some Cisco Spark apps probe for connectivity on port 33434 to send and receive media. While it can be used as a last resort, we suggest that you do not open port 33434 for media. This port is often referred to as the traceroute port and is rate limited by some ISPs. Using this port can cause bad media quality.

Media Node Discovery

Periodically, all Cisco Spark apps perform a media node discovery probe to ensure that the best possible media node is handling the media flow. During this probe, nodes in the Cisco Collaboration Cloud and local hybrid media nodes are discovered.

The Cisco Spark app or endpoint receives a node list from Cisco Collaboration Cloud, performs a ping by sending a STUN (RFC 5389) request, and listens for a response from all the nodes in the list.

The Cisco Spark app or endpoint receives answers from the reachable media nodes. The Cisco Spark app measures the packet's round-trip time (RTT) on UDP, TCP, and TLS based on the send requests and received responses.

The Cisco Spark app reports back to Cisco Collaboration Cloud. The Cisco Spark app reports what media nodes it has connectivity to, which protocol it has connectivity to, and what the measured RTT is. Figure *4* shows how an example of how media node discovery works. This figure is an example only; the real-world placement of these nodes might differ.

Figure 4 Media Node Discovery

After media node discovery occurs, Cisco Collaboration Cloud knows which media nodes can be reached by that particular client or endpoint and on which protocols.

When a meeting starts, Cisco Collaboration Cloud assigns a media node to the Cisco Spark app or endpoint. The assignment is based on information from a previously completed media node discovery probe. Probing is not part of the meeting setup, but is performed when the Cisco Spark app registers to the cloud. The probing data cache expiry is set to 2 hours. The node assignment might not rely just on the RTT measurements, but also on where the other participants in the meeting are joining from.

Media Connectivity

Cisco Spark offers five basic firewall traversal options (as shown in Figure 5):

1) Outbound ANY:ANY
2) STUN inspection
3) Hybrid media node (outbound 1: ANY)
4) IP range whitelisting
5) Fallback (TCP, TLS, HTTP proxy)

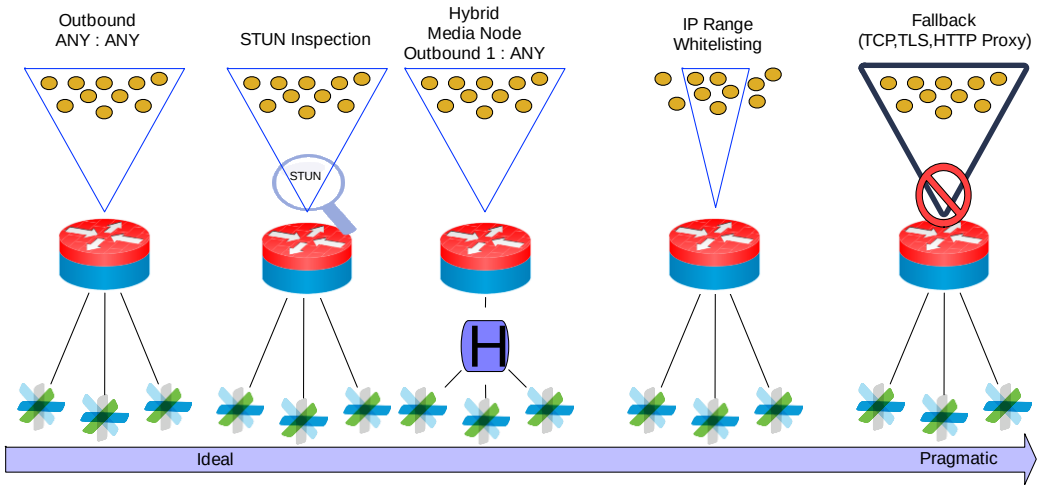

Figure 5 Firewall Traversal Options

The ideal solution is to allow the outbound UDP traffic to ANY. This allows the Cisco Spark app to use the full range of dynamically available media nodes in Cisco Collaboration Cloud. UDP is the preferred protocol for interactive media, because it has lower latency.

ANY:ANY

It's important to understand that Cisco Collaboration Cloud never connects directly to any Cisco Spark app. All needed connections are initiated from the Cisco Spark app. Media flows in both directions using a symmetric inside-initiated, 5-tuple UDP stream outbound to Cisco Collaboration Cloud.[12] The pinhole created in the firewall for this connection is usually closed after 30 seconds if no packets are sent from inside of the network to keep it open.

[12] Sending and receiving media on the same port

STUN Inspection

In theory, STUN inspection adds latency because a firewall needs to run the packet through an inspection engine. However, STUN packets can easily be processed by any decent firewall.

STUN packets are sent as part of the IETF firewall traversal standard, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols," RFC 5245.[13] STUN packets are defined in the IETF standard, "Session Traversal Utilities for NAT (STUN)," RFC 5389. [14]

STUN packets are sent during the call setup as probes. These probes can determine the best possible network path for the media. Those packets contain useful information for both upstream and downstream links. A firewall can use that information to validate any media flow that follows them.

The Cisco ASA can be set up to do a STUN inspection.[6]

Hybrid Media

Hybrid media brings the cloud closer to your network. Using hybrid media is like hosting part of the Cisco Collaboration Cloud in your network that only you have access to.

Hybrid media is delivered as a virtual machine (VM) image (VMware ESXi). See Figure 6. You can install this image anywhere in your network, even behind Network Address Translation (NAT), or in a data center of your choice. The Cisco Spark Hybrid Media Node always initiates any needed connections.

A hybrid media node adds hardware cost because physical CPUs are needed in the local network. For media streams that stay in the local network, this scenario has a significant positive impact on latency, and it also saves bandwidth on the public internet uplink.

[13] https://tools.ietf.org/html/rfc5245
[14] https://tools.ietf.org/html/rfc5389

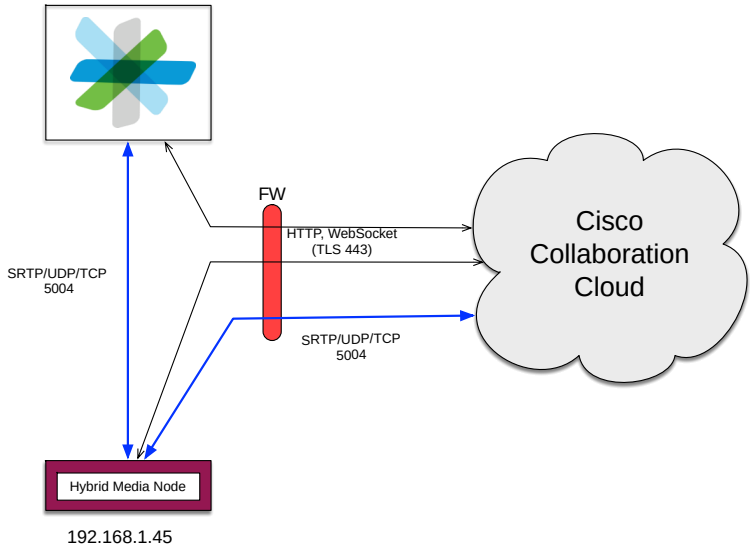

Figure 6 Hybrid Media Ports and Protocols

The Cisco Spark Hybrid Media Node uses the same ports and protocols as the Cisco Spark app when it talks to the cloud.

IP Range Whitelisting

A cloud service is dynamic; it can start up and tear down resources as needed. So that you can get the most benefit from this behavior, we encourage you to open up the firewall for inside-initiated UDP connections to ANY (any IP address) to port 5004 for the best possible media quality.

If you have a security policy that prevents you from opening up inside-initiated connections on UDP to ANY on port 5004, we recommend that you consider if your firewall can support STUN inspections and dynamically open traffic based on STUN packets. Doing so can increase your security while you still can take advantage of all available media nodes.

Another option is to whitelist an IP address range in which Cisco Collaboration Cloud operates media nodes. Bear in mind that whitelisting may reduce media quality, because you can't use a media node that may be closer to you.

Whitelisting specific IP ranges affects the media node discovery because the Cisco Spark app can only reach the media nodes within the whitelisted IP range. This scenario might cause the Cisco Spark app or endpoint to end up with a suboptimal media node.

The Cisco Collaboration Cloud media nodes that operate in the Cisco WebEx data center IP ranges also listen on UDP port 9000. Port 9000 should already be opened during WebEx

deployment.[15] This scenario allows existing WebEx customers to deploy Cisco Spark without any firewall changes. We still recommend that you open UDP port 5004 so that you have better flexibility and the ability to easily distinguish between Cisco WebEx and Cisco Spark traffic in the network.

HTTP Proxy Traversal

The behavior of HTTP proxies can vary. Automatic discovery and configuration are usually done by the Web Proxy Auto-Discovery Protocol (WPAD). WPAD is a DHCP extension that points to a proxy auto-config (PAC) file. The PAC file contains JavaScript that must be executed by the application that wants to traverse the HTTP proxy. The Cisco Spark app relies on the underlying operating system to support this scenario.

Note: Because Android doesn't have this feature, you must manually configure any HTTP proxy on an Android device.

If you configure the underlying operating system to use a proxy, the Cisco Spark app or endpoint can use it when it establishes a connection out to the Cisco Collaboration Cloud. Supported authentication methods can vary between devices and operating systems. See the online Cisco Spark "Network Requirements for Cisco Spark Services" documentation for more detailed information.[16]

Cisco Spark Board, Cisco SX, and Cisco DX video endpoints do not support automatic discovery and configuration; you must manually make these settings on these endpoints.[17] Hard endpoints only support sending the signaling through the HTTP proxy; there is no support for proxying media traffic.

HTTP Proxy in Split Signaling and Media Scenarios

If you configure the Cisco Spark app to use an HTTP proxy, all signaling is sent to the configured proxy address and port. See Figure 7. The Cisco Spark app uses the appropriate proxy signaling (HTTP Connect) to connect to Cisco Collaboration Cloud. Signaling passes through the HTTP proxy and is inspected by the HTTP proxy. To allow the signaling traffic to reach Cisco Collaboration Cloud, it's important that you set the HTTP proxy to allow traffic to the

---

[15] https://cisco-support.webex.com/guest/articles/en_US/Usability_FAQs/WBX264/
[16] https://help.webex.com/docs/DOC-4401

[17] Not yet released.

appropriate URLs. See the Cisco Spark online documentation and the section on URLs for details.[1]

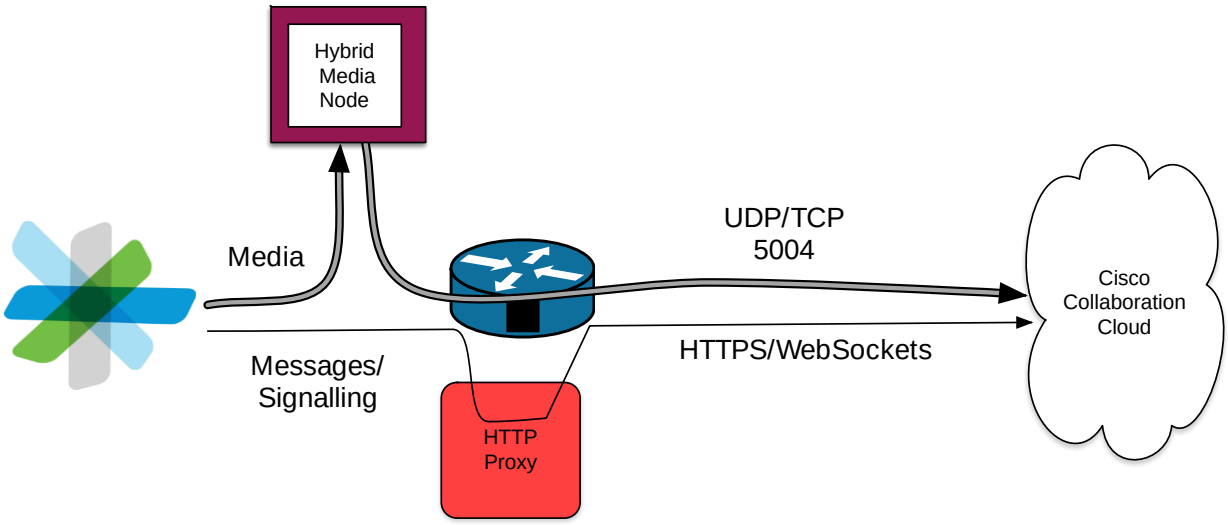

Figure 7 HTTP Proxy and Hybrid Media

If hybrid media is present in the local network, you can use it to handle media. See Figure 8. In this scenario, media no longer needs to traverse the HTTP proxy. This scenario significantly improves the media quality because you can use UDP as the transport protocol. If all the meeting participants are located on the same network, media is not sent to Cisco Collaboration Cloud because the local hybrid media node handles the media streams.

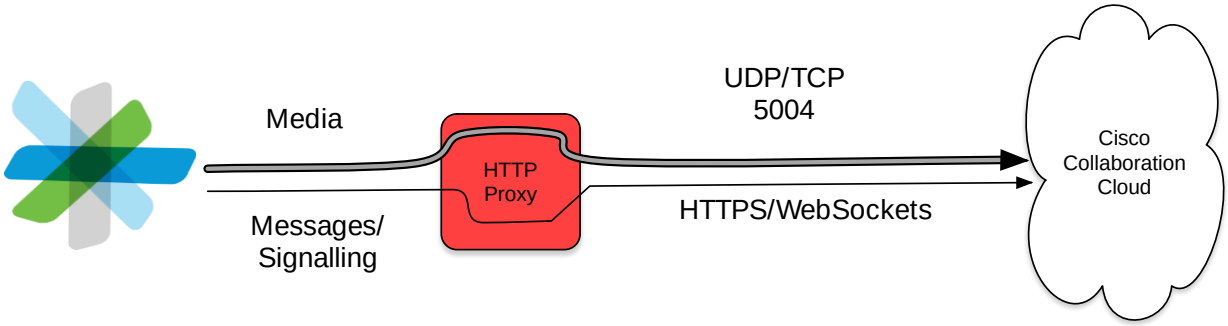

Figure 8 Media Through an HTTP Proxy

If no hybrid media is present in the local network, media can also pass through the HTTP proxy itself. How this is handled by the HTTP proxy depends on the proxy configuration. We do not recommend that you send media through an HTTP proxy because excessive latency is added. An HTTP proxy is not designed to handle interactive audio and video.

We recommend that you consider any of the other firewall traversal alternatives that are described in this white paper. We realize that it might be a major shift in how the enterprise handles internet traffic, but that is the tradeoff using any cloud technology.

Summary

Dealing with enterprise firewall traversal is no easy task, especially when your goal is to ease the adoption of Cisco Spark and minimize the workload for your IT department. Cisco Spark can help you to do the following:

- Initiate outbound connections
  Note: Cisco Collaboration Cloud never initiates any connections to the Cisco Spark apps.
- Use documented ports and protocols to connect to Cisco Collaboration Cloud
- Connect to Cisco owned and controlled URLs for messaging, signaling, and identity management
- Use TCP/TLS fallback for media as an alternate way to connect to Cisco Collaboration Cloud
- Support HTTP proxies to allow the signaling traffic to reach Cisco Collaboration Cloud (with some limitations on hard endpoints)
- Use IP range whitelisting to connect to media nodes within the whitelisted IP range
- Use the Cisco Spark Hybrid Media Node to bring the cloud closer to your network