This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Web Real-Time Communications Deployment Guide

Public Cloud

9/12/2025

Public Cloud

When Genesys servers are hosted on a public cloud, such as Amazon Web Services (AWS), special attention must be paid for handling media paths, since public cloud providers employ both NAT and strict firewall policies.

The firewall policy for the WebRTC Service must allow both HTTP/HTTPS and SRTP.

The firewall policy for the STUN/TURN server must allow the STUN/TURN ports.

Amazon AWS uses a 1-to-1 NAT for all the EC2 instances. Each EC2 instance has a single network interface with a private IP address within the AWS network and each EC2 instance is mapped to a single public IP address that is reachable from the Internet. The 1-to-1 NAT maps the public IP address to the EC2 instance's private IP address. Some STUN server implementations are designed to support hosting behind a 1-1 NAT, so it is possible to also host the STUN server on AWS.

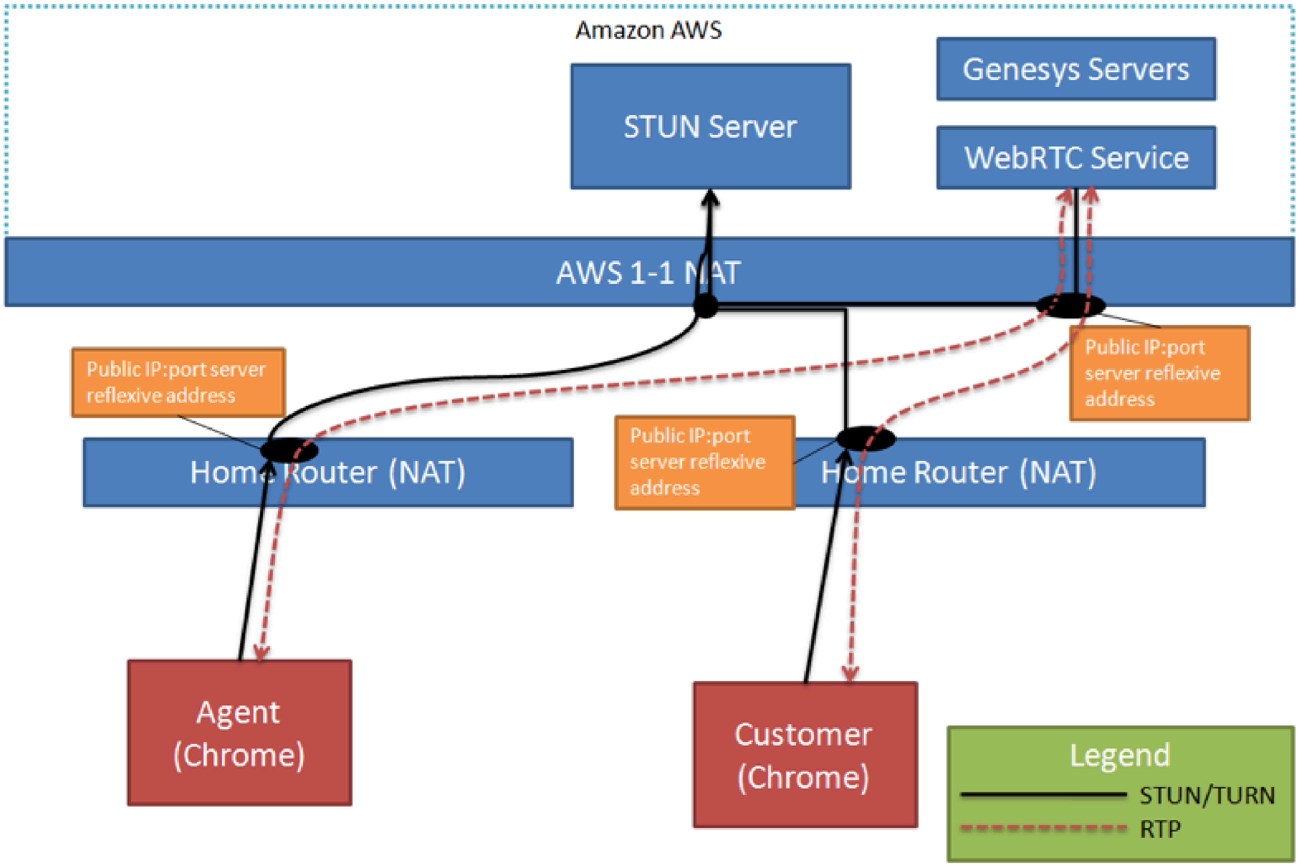
Media servers hosted on AWS must be able to advertise the external IP address in an SDP offer/answer in order to establish the media path with the Genesys WebRTC Service. MCP must be configured to set the external IP address:

```
mpc.sdp.connection = IN IP4 xxx.yyy.zzz.aaa
sip.transport.localaddress = xxx.yyy.zzz.aaa
```

Customer (Browser) – Agent (Browser)

Note: *A web-based agent desktop application implemented using the Genesys WebRTC JavaScript API could be used in this model. Genesys Workspace Web Edition is such a tool that supports audio-only calls, with no video support at this time.*

In this deployment, both customer and agent are using a browser to handle calls. They can discover the server-reflexive address behind their own home router using the STUN server hosted on AWS. The browsers can then establish the media path to the WebRTC Service and have the media path bridged on the WebRTC Service.



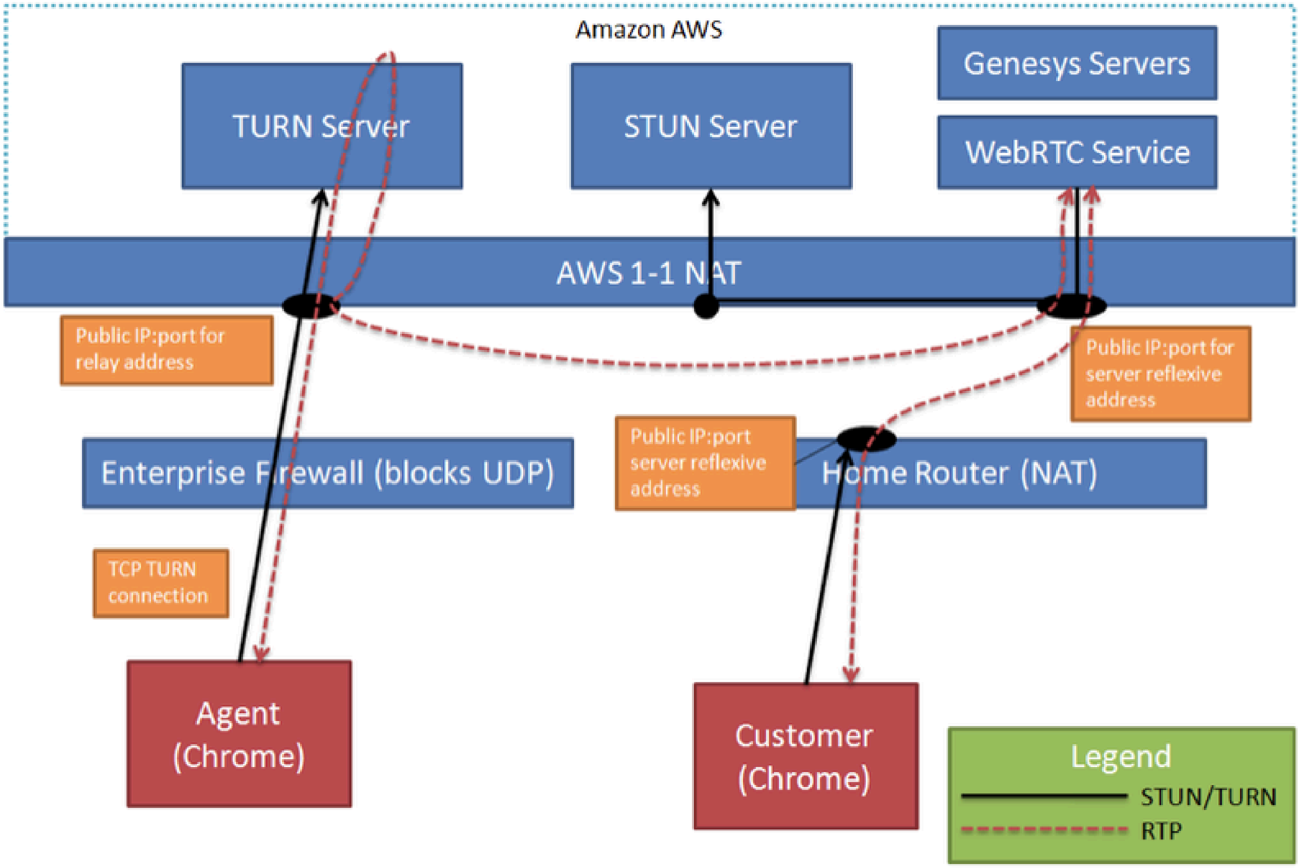
Customer (Browser) – Agent (Browser Behind a Firewall)

Note: *A web-based agent desktop application implemented using the Genesys WebRTC JavaScript API could be used in this model. Genesys Workspace Web Edition is such a tool that supports audio-only calls, with no video support at this time.*

When an agent using a browser is connecting to the public cloud from behind an enterprise firewall, the firewall may block RTP traffic or even block UDP altogether. In this case, the deployment will need to deploy a TURN server on AWS to allow the media path to be relayed through the TURN server.

The TURN server must be opened on a port that is accessible by a browser agent behind the enterprise firewall. For example, if the TURN port is not enabled on the firewall, the TURN server port can be enabled on another port for which the firewall allows UDP or TCP traffic through, such as known HTTP(s) ports.

When the browser agent connects to the WebRTC Service, the browser—acting as a TURN client—reserves TURN relay ports before sending an SDP offer to the WebRTC Service. The media path is relayed through the TURN server to the WebRTC Service and then bridged to the customer browser.


Amazon AWS
TURN Server
STUN Server
Genesys Servers
WebRTC Service
AWS 1-1 NAT
Public IP:port for relay address
Public IP:port for server reflexive address
Public IP:port server reflexive address
Enterprise Firewall (blocks UDP)
Home Router (NAT)
TCP TURN connection
Agent (Chrome)
Customer (Chrome)
Legend
STUN/TURN
RTP