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.

Genesys Interaction Recording API

Using the Screen Recording API

12/14/2025

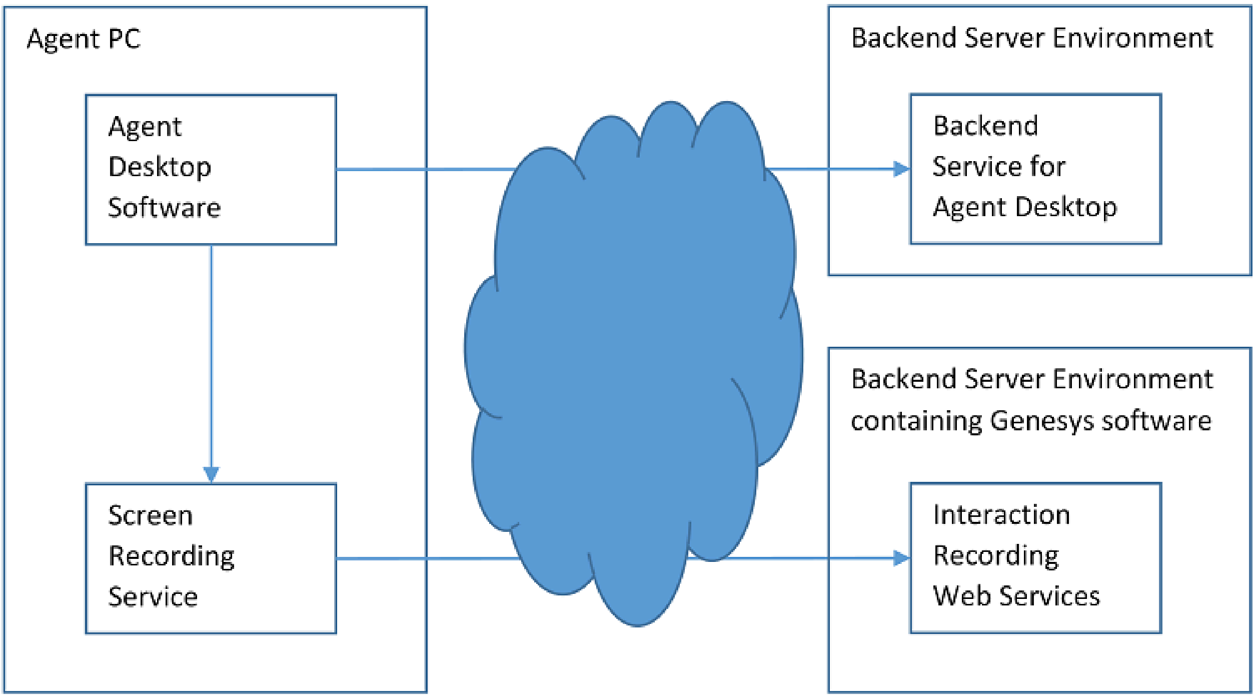
Contents

Using the Screen Recording API

The Screen Recording feature within Genesys Interaction Recording (GIR) allows customers to capture the entire agent screen, including multiple monitors, for both voice and non-voice interactions delivered to the agent desktop. Typically, an agent uses either Workspace Desktop Edition (WDE) or Workspace Web Edition (WWE), which are already integrated with the Screen Recording Service (SRS), as their desktop. However, some customers may have their own custom agent desktop application. This section provides information on how to integrate screen recording functionality into a custom agent desktop.

Components

The following is a simplified component diagram.

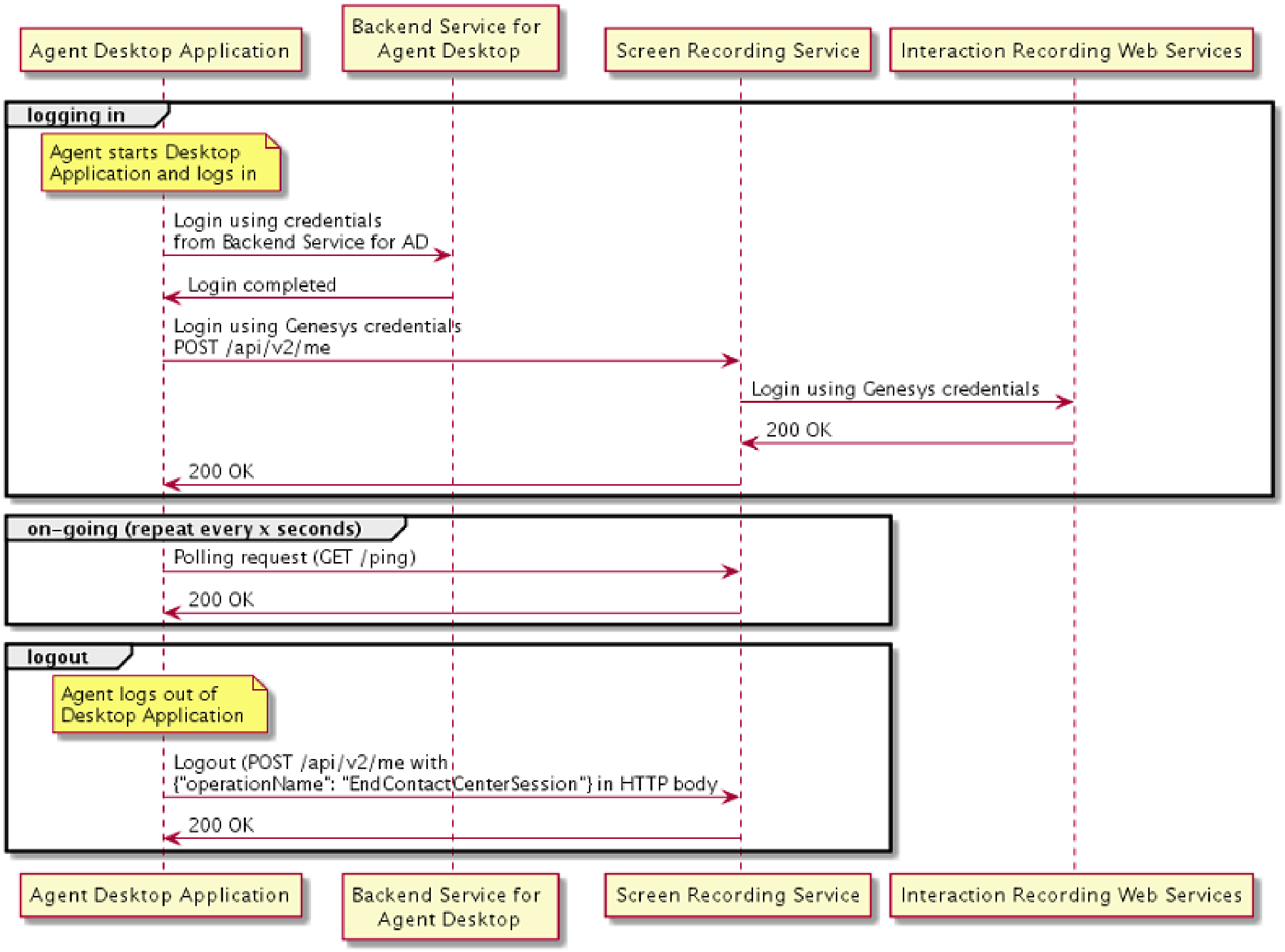

Agent PC
Agent Desktop Software
Screen Recording Service
Backend Server Environment
Backend Service for Agent Desktop
Backend Server Environment containing Genesys software
Interaction Recording Web Services

- **Agent Desktop Software**: The client application running on the agent's workstation. The agent desktop provides users access to interaction information as the interaction is delivered to the agent. This application also provides access to the functions, processes and related applications that are needed to successfully handle a customer interaction.
- **Screen Recording Service (SRS)**: A Windows service that is installed on the agent's machine. SRS receives instructions from the agent desktop and Interaction Recording Web Services (RWS) to control when to start and stop the screen recording on the agent's machine.

- **Interaction Recording Web Services (RWS)**: A software service running in the Genesys environment to control SRS and to store and manage recording files.
- **Backend Service for Agent Desktop**: The software services that the agent desktop application interacts with. Depending on how the software is deployed, it may not be in the same environment as RWS.

Call Flow Diagram

As shown in the figure, the agent desktop application must make appropriate API calls to SRS when the following events occur:


Agent Desktop Application
Backend Service for Agent Desktop
Screen Recording Service
Interaction Recording Web Services
logging in
Agent starts Desktop Application and logs in
Login using credentials from Backend Service for AD
Login completed
Login using Genesys credentials POST /api/v2/me
Login using Genesys credentials
200 OK
200 OK
on-going (repeat every x seconds)
Polling request (GET /ping)
200 OK
logout
Agent logs out of Desktop Application
Logout (POST /api/v2/me with {"operationName": "EndContactCenterSession"} in HTTP body
200 OK
Agent Desktop Application
Backend Service for Agent Desktop
Screen Recording Service
Interaction Recording Web Services

- **Agent login**: When an agent logs into the agent desktop, the Login API must be invoked.
- **Polling**: As the agent is using the agent desktop, the Polling API must be invoked periodically so that the timeout-and-clean-up mechanism is not triggered.
- **Agent logout**: When the agent logs out of the agent desktop, the Logout API must be invoked.

Cross-Origin Resource Sharing (CORS)

If the agent desktop is implemented as a web application running on the browser, this web application makes an HTTP or HTTPS request to the SRS running on the same machine as the browser. Make sure your web application is making requests to SRS as per the W3C CORS standard. Note that based on the CORS specification, if the agent desktop is communicating with its backend services using HTTP, then the request to SRS must also use HTTP. If the agent desktop is communicating with its backend services using HTTPS, then the request to SRS must also use HTTPS.

SRS supports the CORS pre-flight OPTIONS requests.

Cross-Site Request Forgery (CSRF) headers

To prevent Cross-Site Request Forgery (CSRF) attack, the following pair of headers can optionally be used to protect a user from unwittingly terminating a session established with SRS:

```
X-Support-CSRFP
X-CSRF-Token
```

At the time a login request is sent to SRS, if the HTTP request contains the "`X-Support-CSRFP: true`" header, then the HTTP response will contain the "`X-CSRF-Token: <token>`" header with a token that is used only for this session.

Within the same HTTP session, subsequent logout requests must contain the "`X-CSRF-Token: <token>`" request header with the same token initially obtained in the login response header.

A request may be rejected if the HTTP header is not included in the POST request.

Hot Seating vs. Default Place

When an agent logs into the agent desktop, depending on the contact center configuration settings, the agent may be asked to choose the "`place`" where he is located for the login session. This allows calls to be correctly routed to the phone numbers associated with the place at which the agent is located for this login session. This is known as hot seating.

If the agent is not prompted to enter the "`place`" at the time of login, then the default place configured for this agent is always used.

Client Login API

The login API is described in the Login Request with Configuration section.

Hot Seating

When the contact center is using hot seating, the login request payload must contain at least the following parameters.

Parameter	Description
server	The server name and port number, and the protocol (http vs. https) that SRS uses to access RWS. This parameter contains a URL, for example, "https://myserver.com:443". The servers that the URL accesses must be in the same data center as the Genesys software that is working with the backend service for the agent desktop.
place	The name of the place that the agent selected at the time of login. This parameter must match the name of the place object configured in the Genesys configuration or when using Agent Setup.
devices	The names of the DN and the switch objects defined in the Genesys configuration.

Important

If the "place" contains only one DN, then the "devices" attribute is not required. If the "place" contains multiple DNs, then both the "place" and "devices" attributes are required. Every item in the devices list (that is, the "DN" object belonging to the "switch" object) must belong to the specified place.

Example Request Payload

```
[
  {
    "name": "server",
    "value": "https://rws.dc1.example.com"
  },
  {
    "name": "place",
    "value": "alicePlace"
  },
  {
    "name": "devices",
    "value": [
      {
        "dn": "DN1",
        "switch": "SwitchA"
      },
      {
        "dn": "DN2",
        "switch": "SwitchB"
      }
    ]
  }
]
```

> **Important**
>
> If an agent switches place in the middle of an active session, the agent desktop application must send a login API request to SRS again to monitor events from the new place and to control the recording behavior.

Default Place

If a contact center does not use hot seating, the agent will always use the assigned default place. In this case, the login request should contain only the "`server`" parameter.

Parameter	Description
`server`	The server name and port number, and the protocol (http vs. https) that SRS uses to access the RWS. This parameter contains a URL, for example, "https://myserver.com:443". The servers that the URL accesses must be in the same data center as the Genesys software that is working with the backend service for the agent desktop.

Example Request Payload

```
[
  {
     "name": "server",
     "value": "https://rws.dc1.example.com"
  }
]
```

Disaster Recovery

When SRS invokes the login API to access RWS, the "`server`" parameter indicates where SRS should send requests to RWS. The "`peer_server`" parameter should be used to specify the address of the backup RWS in the backup data center. In case of a disaster such as the entire data center becoming unavailable, the agent desktop application may alternatively attempt disaster recovery by sending a login request to SRS with the address of the recovery data center where the backup RWS resides in the "`server`" parameter.

Disaster recovery can be used for both hot seating and default place configurations.

Client Polling API

The polling API is documented in the Client Polling API section.

SRS has a timeout mechanism to terminate screen recordings if it does not receive a polling API call from the agent desktop application. Therefore, while an agent remains logged in to the agent desktop, the agent desktop application must send the polling API periodically to prevent SRS from timing out. The default timeout value in SRS is 60 seconds, so the agent desktop application must invoke the polling API every 30 seconds.

Client Logout API

The logout API is documented in the Logout Request section.

When an agent logs out of the agent desktop, the agent desktop application must send the logout API to SRS. This API uses the same HTTP endpoint as the login API but with a different request payload, as shown below.

```
{"operationName": "EndContactCenterSession"}
```