Record

The `Record` voice element makes a recording of the caller's voice. A prompt is played to the caller then the voice element records the caller's voice until a termination key is inputted, the recording time limit has been reached, or (if the configuration specifies so) the caller hung up. An audio cue (beep) may be activated to signal to the caller that the system is ready to record the caller's voice. Different voice browsers may have varying default maximum lengths for voice recording.

The recording is sent to the Record element by the voice browser and is stored in an audio file in the location specified by the developer. Any pre-existing file with the same name will be overwritten. The element can be configured to produce a non-repeating filename so all recordings can be retained. The format for this filename is audioNR.wav where N is the number of milliseconds since midnight January 1, 1970 (GMT) and R is a random number between 1 to 1000. All recordings are saved in the WAV format.

Settings

Name (Label)	Type	Req'd	Single Setting Value	Substitution Allowed	Default	Notes
noinput_timeout (Noinput Timeout)	string	Yes	`true`	`true`	5s	The maximum time allowed for silence or no keypress before a noinput event is thrown. Possible values are standard time designations including both a non-negative number and a time unit, for example, 3s (for seconds) or 3000ms (for milliseconds). Default = 5s.

max_noinput_count (Max NoInput Count)	int ≥ 0	Yes	`true`	`true`	3	The maximum number of noinput events allowed during input capture. 0 = infinite noinputs allowed.
start_with_beep (Start With Beep)	boolean	Yes	`true`	`true`	`true`	Whether or not to play a beep before recording begins.
terminate_on_dtmf (Terminate On DTMF)	boolean	Yes	`true`	`true`	`true`	Whether or not the caller can end the recording by pressing a touchtone key.
keep_recording_on_hangup (Keep Recording On Hangup)	boolean	Yes	`true`	`true`	`false`	Whether or not the recording is stored if the caller hung up while making the recording. Default = false
max_record_time (Max Record Time)	string	Yes	`true`	`true`	180s	The maximum time (in seconds) the recording is allowed to last. Possible values are standard time designations including a positive integer followed by s (for seconds), for example, 30s. Default = 180s. **Note** Special consideration must be taken for the "ivr record memory session" setting on the gateway and the configured values for the "Max Record Time" settings in the Record element. Depending on the combination of these settings, a caller may exhaust all available memory on the gateway for their session. At which point the gateway will drop the call.

						Note To prevent calls from being dropped while using the Record element, the following formula should be adhered to: "Max Record Time" in seconds * audio codec bitrate in kilobytes/second < "ivr record memory session" setting, in kilobytes. Testing should be done by increasing values for the gateway's "ivr record memory session" setting until an acceptable amount of recorded audio is accepted without exhausting the gateway's session memory (dropped calls).
final_silence (Final Silence)	string	Yes	`true`	`true`	4s	The interval of silence (in seconds or milliseconds) that indicates the end of speech. Possible values are standard time designations including both a positive integer and a time unit identifier, for example, 3s (for 3 seconds) or 300ms (for 300 milliseconds). Default = 4s. **Note** For silence detection to work, you must enable Voice Activity Detection (VAD) in the gateway dial-peers. Manually remove NO VAD from the configuration script and replace it with VAD. **Note** CUBE does not support silence detection.
filename (Filename)	string	No	`true`	`true`	*None*	The filename of the recording (without extension). If left blank, an auto-generated filename will be used.
file_type (File Type)	string enum	Yes	`true`	`true`	`wav`	This specifies the audio type of the file that will hold the recording. Possible values are: `wav` | `vox` | `au` | `other`.

mime_type (Mime Type)	string	Yes	`true`	`true`	*None*	This specifies the MIME type of the file that will hold the recording, if file_type is set to `other`.
file_extension (File Extension)	string	No	`true`	`true`	*None*	This specifies the file extension to use for the recorded file. A file extension different from the file type can be used. For example, with a mime type of *vox*, the file extension could be set to ulaw.
path (Path)	string	No	`true`	`true`	*None*	The path to the file that will hold the recording. Either the path, ftp host, or both must be specified.
ftp_host (FTP Host)	string	No	`true`	`true`	*None*	The domain name of the host to ftp the recording. Either the path, ftp host, or both must be specified.
Secure (Secure)	boolean	Yes	`true`	`true`	`false`	Whether or not to enable Secure File Transfer protocol (SFTP). Default = false, indicates file transfer happens over FTP by default.
ftp_user (FTP User)	string	Yes	`true`	`true`	*None*	The user name to use while FTPing the recording, if ftp_host is set.
ftp_password (FTP Password)	string	Yes	`true`	`true`	*None*	The password to use while FTPing the recording, if ftp_host is set.
ftp_path (FTP Path)	string	No	`true`	`true`	*None*	The directory in which to FTP the recording, if ftp_host is set.
ftp_in_background (FTP In Background)	boolean	Yes	`true`	`true`	`true`	Whether or not the FTP is to be performed in the background, if ftp_host is set.

Note	For recording, use these procedures:
	• Nomatch events cannot be thstrown in this voice element. Since all audio is recorded (except DTMF key presses), there is no reaction on spoken commands (including hotlinks).
	• A noinput event is possible if the voice browser detects no audio once recording has started. If the input timeout has been reached, the noinput event is thrown.
	• The path setting does not require a trailing slash. The voice element will determine the appropriate destination. The path may be specified in operating system specific format (for example, on Windows it might be specified as C:\directory\subdirectory\ and on UNIX it might be */usr/local/directory/*).
	• For a recording to be stored, you can choose either to store it locally or remotely. For locally on the VXML server itself, configure only the filename (`myfile`) and the path (`c:/recordings/`). For remotely on a ftp server, configure the filename (`myfile`) and the FTP details such as: host, user, path, and password. Once your record element is configured, determine the url to access the recording from an external system. Run a simple test by playing the recording from your web browser. Make use of the url: http://<ftpserver>/<ftppath>/filename . Find the correct path to play the audio file and use the same url in the audio element settings.
	• If `terminate_on_DTMF` is *false* or off, recording will stop only after the voice browser reaches the input timeout.
	• Some voice browsers may not accept all options provided for the `file_type` and `mime_type` settings. Check your voice browser documentation for information on supported audio types.
	• It is important to ensure that VXML Server has permission to save audio files to the specified path.
	• VVB silence detection while recording is not supported currently.

Element Data

Name	Type	Notes
filename	string	This stores the filename of the recording (without the path).
filepath	string	This stores the path to the file holding the recording (including the filename).
hungUpWhileRecording	boolean	This stores a *true* if the caller hung up while making the recording, *false* if not.

Exit States

Name	Notes
max_noinput	The maximum number of noinput events has occurred. If the `max_noinput` count is 0, this exit state will never occur.
done	The message was recorded.

Audio Groups

Record Capture

Name (Label)	Req'd	Max1	Notes
initial_audio_group (Initial)	Yes	Yes	Played when the voice element first begins.
noinput_audio_group (No Input)	No	No	Played when a noinput event occurs.

Folder and Class Information

Studio Element Folder Name	Class Name
Record	com.audium.server.voiceElement.record.MRecord

Events

Name (Label)	Notes
Event Handler	You can select either VXML Event or Java Exception as event handler type from the drop-down list.