



Low-Latency Streaming User Guide

# Amazon Interactive Video Service



# Amazon Interactive Video Service: Low-Latency Streaming User Guide

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆或者贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

# Table of Contents

什么是 IVS 低延迟流式传输？ .....	1
延迟 .....	1
全球解决方案、区域控制 .....	2
流传输和查看面向全球 .....	2
控制分区域进行 .....	2
您的通道所在区域 .....	3
IVS 入门 .....	4
步骤 1：创建亚马逊云科技账户 .....	4
步骤 2：设置根用户和管理用户 .....	5
保护您的亚马逊云科技账户根用户 .....	5
创建管理用户 .....	5
步骤 3：设置 IAM 权限 .....	6
使用 IVS 权限的现有策略 .....	6
可选：为 Amazon IVS 权限创建自定义策略 .....	6
创建新用户并添加权限 .....	8
向现有用户添加权限 .....	9
步骤 4：使用可选录制创建通道 .....	9
自动录制到 Amazon S3 .....	10
控制台说明 .....	11
CLI 说明 .....	18
步骤 5：设置流式传输软件 .....	22
使用 Amazon IVS 广播开发工具包进行流传输 .....	22
使用 Amazon IVS 控制台进行流式传输 .....	23
通过 OBS Studio 进行流传输 .....	23
使用 FFmpeg 对录制的视频进行流传输 .....	25
步骤 6：查看您的实时流 .....	26
步骤 7：查看您的服务限额限制（可选） .....	26
第 8 步：阻止不想要的内容和浏览者（推荐） .....	27
控制台说明（播放限制政策） .....	27
CLI 说明（播放限制政策） .....	28
如何禁用录制 .....	28
在 IVS 流中启用多台主机 .....	29
开始使用 .....	29
控制台说明 .....	29

CLI 说明 .....	30
广播舞台：客户端与服务器端合成 .....	31
演示 .....	32
1. 创建舞台 .....	33
2. 分发参与者令牌 .....	35
3. 加入舞台 .....	35
4. 广播舞台 .....	38
监控 .....	41
先决条件 .....	41
访问流会话数据 .....	42
控制台说明 .....	42
AWS SDK 说明 .....	42
CLI 说明 .....	43
按运行状况筛选流 .....	45
控制台说明 .....	45
CLI 说明 .....	46
ConcurrentStreams 的 CloudWatch 运行状况维度 .....	46
访问 CloudWatch 指标 .....	46
CloudWatch 控制台说明 .....	46
CLI 说明 .....	47
CloudWatch 指标：IVS 低延迟直播功能。 .....	47
IVS 广播 SDK .....	52
平台要求 .....	52
本机平台 .....	52
桌面浏览器 .....	53
移动浏览器 .....	53
Webviews .....	53
所需设备访问 .....	53
支持 .....	54
版本控制 .....	54
Web 指南 .....	55
.....	55
开始使用 .....	55
已知问题和解决方法 .....	61
Android 指南 .....	63
开始使用 .....	64



高级使用案例 .....	67
已知问题和解决方法 .....	72
iOS 指南 .....	73
开始使用 .....	74
高级使用案例 .....	79
iOS 如何选择相机分辨率和帧率 .....	87
已知问题和解决方法 .....	87
混合器指南 .....	89
术语 .....	89
画布属性 .....	91
插槽属性 .....	91
配置广播会话以进行混合 .....	94
添加槽 .....	97
移除插槽 .....	97
动画转换 .....	97
镜像广播 .....	99
自定义图像源 .....	101
Android .....	101
iOS .....	102
IVS 播放器开发工具包 .....	104
浏览器和平台要求 .....	105
桌面浏览器 .....	105
移动浏览器 .....	106
本机平台 .....	107
减少第三方播放器的延迟 .....	107
iOS Safari .....	107
纯音频播放 .....	108
支持 .....	108
版本控制 .....	108
Web 指南 .....	109
开始使用 .....	109
框架集成 .....	112
使用内容安全策略 .....	113
已知问题和解决方法 .....	113
Android 指南 .....	114
开始使用 .....	115

开发工具包大小 .....	118
已知问题和解决方法 .....	118
iOS 指南 .....	119
开始使用 .....	119
开发工具包大小 .....	125
已知问题和解决方法 .....	125
Video.js 集成 .....	126
开始使用 .....	126
事件 .....	129
错误 .....	129
插件 .....	130
内容安全策略 .....	131
函数 .....	131
currentTime .....	132
dispose .....	133
duration .....	134
getIVSEvents .....	134
getIVSPlayer .....	135
load .....	135
play .....	136
playbackRate .....	136
seekable .....	137
JW 播放器集成 .....	137
开始使用 .....	138
事件 .....	139
错误 .....	140
内容安全策略 .....	141
限制 .....	141
将元数据嵌入视频流中 .....	142
什么是定时元数据？ .....	142
设置 IAM 权限 .....	142
插入定时元数据 .....	143
使用 Amazon CLI .....	143
使用 Amazon IVS API .....	144
使用 IVS 广播 SDK .....	144
使用定时元数据 .....	144

示例演示：测验应用程序 .....	145
查看定时元数据 .....	146
了解更多信息 .....	146
设置私有通道 .....	147
私有通道的工作流 .....	147
创建或导入播放密钥 .....	148
创建新的密钥对 .....	148
导入现有公有密钥 .....	149
API 请求 .....	150
在通道上启用播放授权 .....	150
控制台说明 .....	150
CLI 说明 .....	151
API 请求（创建与更新） .....	152
生成和签名播放令牌 .....	152
令牌架构 .....	152
说明 .....	154
列出播放密钥 .....	154
控制台说明 .....	154
CLI 说明 .....	155
API 请求 .....	155
删除播放密钥 .....	156
控制台说明 .....	156
CLI 说明 .....	156
API 请求 .....	156
获取有关播放密钥的信息 .....	157
控制台说明 .....	157
CLI 说明 .....	157
API 请求 .....	157
撤消查看者会话 .....	158
CLI 说明 .....	158
API 请求 .....	158
自动录制到 Amazon S3 .....	159
S3 前缀 .....	159
录制内容 .....	160
字节范围播放列表 .....	161
缩略图 .....	161

合并片段化的流 .....	161
资格 .....	162
已知问题 .....	162
JSON 元数据文件 .....	162
示例：recording_started.json .....	166
示例：recording_ended.json .....	167
示例：recording_failed.json .....	168
查找录制的呈现 .....	170
播放私有存储桶中的录制内容 .....	172
亚马逊 CloudFront 配送 .....	172
从亚马逊播放 CloudFront .....	172
将 Amazon EventBridge 与 IVS 配合使用 .....	174
为亚马逊 IVS 创建亚马逊 EventBridge 规则 .....	176
示例：流状态更改 .....	177
示例：流运行状况更改 .....	178
示例：违反限制 .....	179
示例：录制状态更改 .....	182
使用 AWS CloudTrail 记录 IVS API 调用 .....	186
CloudTrail 中的 Amazon IVS 信息 .....	186
了解 Amazon IVS 日志文件条目 .....	187
安全性 .....	189
数据保护 .....	189
身份和访问管理 .....	190
受众 .....	190
Amazon IVS 如何与 IAM 配合使用 .....	191
身份 .....	191
策略 .....	191
基于 Amazon IVS 标签的授权 .....	192
角色 .....	193
特权访问和非特权访问 .....	193
使用策略的最佳实践 .....	193
基于身份的策略示例 .....	193
故障排除 .....	198
Amazon IVS 的托管式策略 .....	199
IVSReadOnlyAccess .....	200
IVSFullAccess .....	200

策略更新 .....	200
使用服务相关角色 .....	201
Amazon IVS 的服务相关角色权限 .....	201
为 Amazon IVS 创建服务相关角色 .....	202
为 Amazon IVS 编辑服务相关角色 .....	202
删除适用于 Amazon IVS 的服务相关角色 .....	202
Amazon IVS 服务相关角色支持的区域 .....	203
日志记录和监控 .....	203
事件响应 .....	203
恢复能力 .....	203
Amazon IVS 视频数据层面 .....	204
基础设施安全性 .....	204
API 调用 .....	204
流式处理和播放 .....	205
服务限额 .....	206
服务配额增加 .....	206
API 调用速率限额 .....	206
其他限额 .....	208
Service Quotas 与 CloudWatch 使用量指标集成 .....	211
为使用情况指标创建 CloudWatch 警报 .....	212
流传输配置 .....	214
先决条件 .....	214
降低延迟 .....	214
避免使用第三方流传输/转发服务 .....	215
编码器设置 .....	215
流摄取：编解码器、RTMPS 和端口 443 .....	215
分辨率/比特率/FPS .....	216
通道类型 .....	216
视频设置 .....	222
音频设置 .....	222
使用 CBR，而不是 VBR .....	222
使用渐进式信号 .....	222
网络要求 .....	223
隐藏式字幕 .....	223
使用 FFmpeg 的流 .....	224
使用 Amazon IVS 广播 SDK 的流 .....	225

测试视频流 .....	225
问题排查常见问题 .....	226
广播和编码 .....	226
什么是流匮乏？ .....	227
为什么流突然停止了？ .....	227
当我在流式传输时切换网络后会发生什么？ .....	228
如何通过 IVS 实现多区域冗余？ .....	228
如何对 IVS Web 广播 SDK 会话进行问题排查？ .....	229
如何使用 Google Chrome 的 WebRTC-internals 指标来评估 IVS Web 广播 SDK 会话？ .....	230
监控和事件 .....	231
如何监控流匮乏事件？ .....	231
如何使用 Amazon CloudWatch 来监控 IVS 服务配额？ .....	231
如何使用 IVS 流运行状况诊断流不稳定的问题？ .....	232
流回放 .....	240
如何调试 IVS 回放器的行为？ .....	240
为什么所有观众的回放都冻结/停止？ .....	240
是什么导致了 IVS 回放器缓冲？ .....	240
自动录制到 Amazon S3 .....	241
为什么会缺失部分录制内容？ .....	241
KMS-S3 加密能否与自动录制到 S3 一起使用？ .....	242
其他主题 .....	242
“待验证”错误是什么意思？ .....	242
我能否估计使用 IVS 的成本？ .....	242
不想要的内容和浏览者 .....	243
正在检测非预期内容 .....	243
异常检测 .....	243
自定义内容审核 .....	244
防止不想要的内容和浏览者 .....	244
停止流并重置流密钥 .....	244
使用私有通道 .....	245
使用播放限制政策 .....	245
成本 .....	246
实时视频 .....	246
自动录制到 Amazon S3 .....	246
存储录制视频 .....	247
提供录制视频 .....	250

资源与支持 .....	251
资源 .....	251
演示 .....	251
合作伙伴解决方案 .....	251
分析 .....	252
交互性 .....	252
面部和背景滤镜 .....	252
支持 .....	253
术语表 .....	254
文档历史记录 .....	268
Low-Latency Streaming User Guide 变更 .....	268
IVS Low-Latency Streaming API Reference 更改 .....	335
舞台 API 参考更改 .....	341
IVS Chat API 文档更改 .....	341
发布说明 .....	344
2024年2月1日 .....	344
亚马逊 IVS Broadcast SDK : 安卓 1.14.1、iOS 1.14.1、Web 1.8.0 ( 低延迟直播 ) .....	344
2024 年 1 月 31 日 .....	345
无令牌播放限制 .....	345
2024 年 1 月 25 日 .....	345
纯音频播放 .....	345
2024 年 1 月 18 日 .....	345
亚马逊 IVS Player SDK 1.24.0 .....	345
2024 年 1 月 3 日 .....	347
亚马逊 IVS Broadcast SDK : 安卓 1.13.4、iOS 1.13.4、Web 1.7.0 ( 低延迟直播 ) .....	347
2023 年 12 月 4 日 .....	348
Amazon IVS 广播 SDK : Android 1.13.2 和 iOS 1.13.2 ( 低延迟直播 ) .....	348
2023 年 11 月 21 日 .....	349
Amazon IVS 广播 SDK : Android 1.13.1 ( 低延迟直播 ) .....	349
2023 年 11 月 17 日 .....	350
Amazon IVS 广播 SDK : Android 1.13.0 和 iOS 1.13.0 ( 低延迟直播 ) .....	350
2023 年 11 月 14 日 .....	352
Amazon IVS 播放器 SDK 1.23.0 .....	352
2023 年 10 月 16 日 .....	353
Amazon IVS 广播 SDK : Web 1.6.0 ( 低延迟直播 ) .....	353
2023 年 10 月 12 日 .....	354

Amazon IVS 广播 SDK : Android 1.12.1 ( 低延迟直播 )	354
2023 年 10 月 3 日	354
Amazon IVS 播放器 SDK 1.22.0	354
2023 年 10 月 2 日	356
控制台内流式传输	356
2023 年 9 月 14 日	356
Amazon IVS 广播 SDK : Web 1.5.2 ( 低延迟流式传输 )	356
2023 年 8 月 23 日	357
Amazon IVS 广播 SDK : Web 1.5.1、安卓 1.12.0 和 iOS 1.12.0 ( 低延迟流式传输 )	357
2023 年 8 月 23 日	358
Amazon IVS 广播 SDK : Android 1.7.6 ( 低延迟流式传输 )	358
2023 年 8 月 22 日	359
Amazon IVS 回放器 SDK 1.21.0	359
2023 年 8 月 7 日	360
Amazon IVS 广播 SDK : Web 1.5.0、Android 1.11.0 和 iOS 1.11.0	360
2023 年 7 月 17 日	361
R2S3 渲染筛选和缩略图增强功能	361
2023 年 7 月 14 日	362
Amazon IVS 回放器 SDK 1.20.0	362
2023 年 7 月 13 日	364
Amazon IVS 广播 SDK : Web 1.4.0、Android 1.10.0 和 iOS 1.10.0	364
2023 年 6 月 28 日	366
私有通道的查看者会话撤消	366
2023 年 6 月 27 日	366
Amazon IVS 广播 SDK : iOS 1.9.1	366
2023 年 6 月 27 日	367
Amazon IVS 广播 SDK 1.7.5	367
2023 年 6 月 16 日	368
Amazon IVS 广播 SDK : Web 1.3.3	368
2023 年 6 月 2 日	369
高级通道类型	369
2023 年 6 月 1 日	369
Amazon IVS 广播 SDK : Android 1.9.0 和 iOS 1.9.0	369
2023 年 5 月 23 日	372
Amazon IVS 回放器 SDK 1.19.0	372
2023 年 5 月 16 日	374



Amazon IVS 广播 SDK : iOS 1.8.1 .....	374
2023 年 5 月 16 日 .....	375
Amazon IVS 广播 SDK 1.7.4 .....	375
2023 年 5 月 11 日 .....	375
多台主机运行状况 .....	375
2023 年 5 月 1 日 .....	376
Amazon IVS Web 广播 SDK 1.3.2 .....	376
2023 年 4 月 27 日 .....	376
增加舞台参与者 .....	376
2023 年 4 月 4 日 .....	377
Amazon IVS 回放器 SDK 1.18.0 .....	377
2023 年 3 月 30 日 .....	378
RTMP 支持 .....	378
2023 年 3 月 29 日 .....	379
私人频道的一次性令牌 .....	379
2023 年 3 月 28 日 .....	379
Amazon IVS Web 广播 SDK 1.3.1 .....	379
2023 年 3 月 23 日 .....	379
在流上支持多台主机 ( 舞台资源 ) .....	379
2023 年 3 月 23 日 .....	379
Amazon IVS 广播 SDK : Android 1.8.0、iOS 1.8.0 和 Web 1.3.0 .....	379
2023 年 3 月 2 日 .....	382
Amazon IVS 广播 SDK : Android 1.7.3 .....	382
2023 年 2 月 28 日 .....	382
Amazon IVS 回放器 SDK 1.17.0 .....	382
2023 年 2 月 16 日 .....	384
用于自动录制到 S3 的字节范围标签和清单文件 .....	384
2023 年 1 月 31 日 .....	384
Amazon IVS Chat 客户端消息收发 SDK : Android 1.1.0 .....	384
2023 年 1 月 17 日 .....	385
Amazon IVS 回放器 SDK 1.16.0 .....	385
2022 年 12 月 9 日 .....	386
向自动记录到 S3 清单文件添加了时间戳 .....	386
2022 年 12 月 6 日 .....	386
Amazon IVS 广播 SDK : Android 1.7.2 .....	386
2022 年 11 月 17 日 .....	387

聊天记录 .....	387
2022 年 11 月 9 日 .....	388
亚马逊 IVS 聊天客户端消息传送软件开发工具包 : 1.0.2 JavaScript .....	388
2022 年 11 月 1 日 .....	388
Amazon IVS 播放器 SDK 1.14.0 .....	388
2022 年 10 月 18 日 .....	390
亚马逊 IVS 聊天客户端消息传送软件开发工具包 : 1.0.1 JavaScript .....	390
2022 年 10 月 6 日 .....	390
Amazon IVS 广播 SDK 1.7.1 .....	390
2022 年 9 月 22 日 .....	391
Amazon IVS 广播 SDK 1.7.0 .....	391
2022 年 9 月 20 日 .....	393
Amazon IVS 播放器 SDK 1.13.0 .....	393
2022 年 9 月 15 日 .....	395
竖屏视频改进 ( 最终版本 ) .....	395
2022 年 9 月 12 日 .....	395
Amazon IVS 广播 SDK 1.5.2 : iOS .....	395
2022 年 9 月 8 日 .....	396
Amazon IVS Chat 客户端消息收发 SDK : Android 1.0.0 和 iOS 1.0.0 .....	396
2022 年 9 月 2 日 .....	397
Amazon IVS Web 广播 SDK 1.2.0 .....	397
2022 年 8 月 30 日 .....	397
合并片段化的流 .....	397
2022 年 8 月 9 日 .....	398
Amazon IVS Web 播放器 SDK 1.12.0 .....	398
2022 年 7 月 28 日 .....	398
Amazon IVS iOS 广播 SDK 1.5.1 .....	398
2022 年 7 月 21 日 .....	399
Amazon IVS Web 广播 SDK .....	399
2022 年 7 月 14 日 .....	399
Amazon IVS iOS 播放器 SDK 1.8.3 .....	399
2022 年 6 月 28 日 .....	400
Amazon IVS 播放器 Web SDK 1.11.0 .....	400
2022 年 6 月 22 日 .....	401
Amazon IVS 广播 SDK 1.5.0 .....	401
2022 年 6 月 9 日 .....	403

垂直视频改进 .....	403
2022 年 5 月 24 日 .....	404
Amazon IVS Web 和 Android 播放器开发工具包 1.10.0 .....	404
2022 年 4 月 28 日 .....	406
流运行状况更新 .....	406
2022 年 4 月 26 日 .....	406
Amazon IVS Chat .....	406
2022 年 4 月 22 日 .....	407
Amazon IVS iOS 播放器 SDK 1.8.2 .....	407
2022 年 4 月 19 日 .....	408
Amazon IVS 广播 SDK 1.4.0 .....	408
2022 年 3 月 31 日 .....	409
Amazon IVS iOS 播放器 SDK 1.8.1 .....	409
2022 年 3 月 3 日 .....	410
Amazon IVS 广播 SDK 1.3.0 .....	410
2022 年 3 月 1 日 .....	412
Amazon IVS 播放器 SDK 1.8.0 .....	412
2022 年 2 月 3 日 .....	414
Amazon IVS 广播 SDK : Android 1.2.1 .....	414
2022 年 1 月 20 日 .....	415
Amazon IVS 播放器开发工具包 1.7.0 .....	415
2022 年 1 月 18 日 .....	418
R2S3 缩略图配置 .....	418
2021 年 12 月 9 日 .....	418
Amazon IVS 广播开发工具包 1.2.0 .....	418
2021 年 11 月 23 日 .....	421
Amazon IVS 播放器开发工具包 1.6 .....	421
2021 年 11 月 18 日 .....	423
流运行状况 .....	423
2021 年 10 月 20 日 .....	424
Amazon IVS 广播开发工具包 1.1.0 : Android 和 iOS .....	424
2021 年 9 月 29 日 .....	427
Amazon IVS 播放器开发工具包 : Android 1.5.1 .....	427
2021 年 9 月 28 日 .....	428
Amazon IVS 播放器开发工具包 1.5.0 .....	428
2021 年 9 月 8 日 .....	431

Amazon IVS 播放器软件开发工具包 1.4.1 .....	431
2021 年 8 月 13 日 .....	432
ListTagsForResource API 端点 .....	432
2021 年 8 月 10 日 .....	432
Amazon IVS 播放器开发工具包 1.4.0 .....	432
2021 年 7 月 27 日 .....	437
Amazon IVS 广播开发工具包 : Android 1.0.0 和 iOS 1.0.0 .....	437
2021 年 6 月 1 日 .....	438
Amazon IVS 播放器开发工具包 : Android 1.3.3 和 iOS 1.3.3 .....	438
2021 年 5 月 19 日 .....	438
Amazon IVS 播放器开发工具包 : Android 1.3.2 .....	438
2021 年 5 月 5 日 .....	439
Amazon IVS 播放器开发工具包 1.3 .....	439
2021 年 4 月 26 日 .....	442
Service Quotas 与 CloudWatch 使用量指标集成 .....	442
2021 年 4 月 13 日 .....	443
新 CloudWatch 指标 .....	443
2021 年 4 月 7 日 .....	443
自动记录到 S3 (R2S3) .....	443
2021 年 1 月 28 日 .....	443
Amazon IVS 播放器开发工具包 : JW 播放器集成 1.2.0 .....	443
2020 年 12 月 16 日 .....	444
Amazon IVS 回放器 : SDK for Android 1.2.1 .....	444
2020 年 11 月 23 日 .....	444
Amazon IVS 播放器开发工具包 1.2.0 .....	444
2020 年 11 月 12 日 .....	446
新增事件字段 stream_id .....	446
2020 年 11 月 9 日 .....	446
在控制台添加元数据查看功能 .....	446
2020 年 10 月 30 日 .....	446
CloudFormation Support .....	446
2020 年 10 月 27 日 .....	446
提高通道、CCV 和 CCB 的数量限制 .....	446
2020 年 10 月 9 日 .....	447
全新 Service Quotas 和 EventBridge 活动 .....	447
Amazon IVS 播放器 : 适用于 Web 的开发工具包 1.1.2 .....	447

---

2020 年 10 月 7 日 .....	447
Amazon IVS 播放器开发工具包 1.1.0 .....	447
2020 年 9 月 14 日 .....	450
新增事件字段 channel_name .....	450
2020 年 8 月 19 日 .....	450
播放授权 ( 私有通道 ) .....	450
2020 年 8 月 11 日 .....	450
Amazon IVS 播放器 : SDK for iOS 1.0.6 .....	450
2020 年 8 月 5 日 .....	450
将亚马逊 EventBridge 与亚马逊 IVS 配合使用 .....	450
2020 年 7 月 15 日 .....	450
播放器版本 1.0 .....	451
.....	cdliii

# 什么是 Amazon IVS 低延迟流式传输？

Amazon Interactive Video Service (IVS) 是一项托管直播视频流服务，支持您：

- 创建通道并在几分钟内开始流传输。
- 在 ultra-low-latency 直播视频的同时，打造引人入胜的互动体验。
- 将视频大规模分发到一系列设备和平台。
- 轻松集成到网站和应用程序中。

Amazon IVS 让您专注于构建自己的交互式应用程序和受众体验。借助 Amazon IVS，您无需管理基础设施或开发和配置视频工作流的组件，从而确保安全、可靠和经济高效。

Amazon IVS 支持 RTMPS 和 RTMP 流式传输。RTMPS 是通过 TLS 运行的 RTMP ( 实时消息传递协议 ) 的安全版本。RTMP 是通过网络传输视频的行业标准。

除此处的产品文档外，请参阅 <https://ivs.rocks/>，此网站专用于浏览已发布内容 ( 演示、代码示例、博客文章 )、估算成本并通过现场演示体验 Amazon IVS。

## 延迟

延迟是从摄像头捕获实时流到流显示在观看者屏幕上的延迟。Amazon IVS 具有可以提供视频的功能，如下所示：

- 超低延迟：Amazon IVS 通道可以提供延迟在 5 秒以下的视频。
- 实时延迟：IVS 舞台可以提供延迟在 300 毫秒以下的视频。舞台上的所有参与者都会体验到这种增强的“实时延迟”。( 请注意，如果将舞台广播到 IVS 频道，则通道观众将获得超低延迟。 )

对于传统的过顶 (OTT) 流，延迟可能高达 30 秒。

低延迟是构建良好的互动式用户体验以丰富受众体验的关键组成部分。它允许主播、品牌和社区以直接而个性化的方式与现场受众建立联系。

观察到的延迟可能因用户而异，原因是：

- 主播和观看者的地理位置。
- 网络类型和速度。

- 流式处理链中的单个组件。
- 流式处理协议和输出格式。

有关更多信息，请参阅 Amazon IVS 流式处理配置中的[降低延迟](#)。

## 全球解决方案、区域控制

### 流传输和查看面向全球

您可以使用 Amazon IVS 向全球的查看者进行流传输：

- 当您进行流传输时，Amazon IVS 会自动在您附近的位置提取视频。
- 查看者可以通过 Amazon IVS 内容分发网络在全球范围内观看您的实时流。

换一种说法是，“数据层面”是全球性的。数据层面是指流传输/提取和查看。

### 控制分区域进行

虽然 Amazon IVS 数据层面是全球性的，但“控制层面”是区域性的。控制层面是指 Amazon IVS 控制台、API 和资源（通道、流密钥、播放密钥对和录制配置）。

换句话说，Amazon IVS 是一种“区域性 AWS 服务”。即每个区域中的 Amazon IVS 资源都独立于其他区域中的类似资源。例如，您在一个区域中创建的通道与您在其他区域中创建的通道无关。

当您使用资源（例如，创建通道）时，您必须指定创建资源的区域。随后，当您管理资源时，您必须从创建资源的同一区域执行此操作。

如果您使用的是...	您可以通过以下方式指定区域...
Amazon IVS 控制台	使用导航栏右上角的 Select a Region (选择区域) 下拉菜单。
Amazon IVS API	使用合适的服务终端节点。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a> 。  ( 如果您通过开发工具包访问 API，请设置开发工具包的 region 参数。请参阅 <a href="#">用于在 AWS 上进行构建的工具</a> 。 )

如果您使用的是...	您可以通过以下方式指定区域...
AWS CLI	或者： <ul style="list-style-type: none"><li>• 将 <code>--region &lt;aws-region&gt;</code> 附加到您的 CLI 命令。</li><li>• 将区域放入您的本地 AWS 配置文件中。</li></ul>

请记住，无论在哪个区域创建通道，您都可以从任何地方流传输到 Amazon IVS，从而使查看者可以从任何地方观看。

## 您的通道所在区域

您的通道区域是创建通道时分配的 ARN ( Amazon 资源名称 ) 的一部分。在您创建通道时：

- Amazon IVS 控制台将 ARN 显示在页面的 General configuration (常规配置) 区域。随后，控制台将始终在右上角显示您的区域 ( 位置 )。
- Amazon IVS API 将在通道对象的 `arn` 字段中返回 ARN。



# IVS 低延迟流式传输入门

本文档将引导您完成设置第一个 Amazon Interactive Video Service (IVS) 实时流的步骤。

## 主题

- [步骤 1：创建亚马逊云科技账户](#)
- [步骤 2：设置根用户和管理用户](#)
- [步骤 3：设置 IAM 权限](#)
- [步骤 4：使用可选录制创建通道](#)
- [步骤 5：设置流式传输软件](#)
- [步骤 6：查看您的实时流](#)
- [步骤 7：查看您的服务限额限制（可选）](#)
- [第 8 步：阻止不想要的内容和浏览者（推荐）](#)
- [如何禁用录制](#)

## 步骤 1：创建亚马逊云科技账户

要使用 Amazon IVS，您需要一个亚马逊云科技账户。如果您还没有账户，系统会在您注册时提示您创建一个账户。创建亚马逊云科技账户：

1. 打开 <https://portal.aws.amazon.com/billing/signup>。
2. 按照屏幕上的说明进行操作。

在注册时，您将接到一通电话或收到一条短信，要求您输入一个验证码。此外，您还必须提供账单信息，尽管基本服务套餐是免费的。如果您没有使用注册的任何亚马逊云科技服务，系统将不会针对它们向您收费。

3. 创建账户后，您将收到一封电子邮件，其中包含登录 URL 和用户名以及另一封带有登录密码的电子邮件（来自您的 Amazon 账户管理员）。首次登录时，必须更改密码。

如果您想要使用现有的亚马逊云科技账户，请确保该账户使用 Amazon IVS 支持的亚马逊云科技区域：

1. 导航到 [Amazon IVS Console](#) ( Amazon IVS 控制台 )。如果您看到正常的 IVS 控制台页面 ( 显示“全球解决方案，区域内容” )，则没有问题；请跳到[步骤 2：设置根用户和管理用户](#)。如果您被重新定向到 Amazon“不支持的区域”页面，则需要选择一个新区域。
2. 选择相应的选项卡 ( 直播，用于 IVS；直播聊天，用于 IVS 聊天 )，然后选择列出的区域之一。请记住您选择了哪个区域；稍后您会需要它。

在任何时候，您都可以通过转至 <https://aws.amazon.com/> 并选择 My Account ( 我的账户 ) 来查看亚马逊云科技账户活动并管理您的账户。

## 步骤 2：设置根用户和管理用户

当您注册亚马逊云科技账户时，系统会创建一个亚马逊云科技账户根用户。根用户有权访问该账户中的所有亚马逊云科技服务和资源。作为安全最佳实践，请[为管理用户分配管理访问权限](#)，并且只使用根用户执行[需要根用户访问权限的任务](#)。

### 保护您的亚马逊云科技账户根用户

1. 要以管理用户身份登录 IAM Identity Center，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录 URL。要获取使用 IAM Identity Center 用户登录方面的帮助，请参阅《Amazon 登录用户指南》中的[登录 Amazon 访问门户](#)。

要获取使用根用户登录方面的帮助，请参阅《Amazon 登录用户指南》中的[以根用户身份登录](#)。

2. 对您的根用户启用多重身份验证 ( MFA )。

有关说明，请参阅《IAM 用户指南》中的[为您的亚马逊云科技账户根用户启用虚拟 MFA 设备 \( 控制台 \)](#)。

### 创建管理用户

您应该创建一个管理用户，以避免使用根用户执行日常任务。

- 对于日常管理任务，请在 AWS IAM Identity Center ( Amazon Single Sign-On 的后继者 ) 中为管理用户分配管理访问权限。有关说明，请参阅《AWS IAM Identity Center ( Amazon Single Sign-On 的后继者 ) 用户指南》中的[入门](#)。
- 要以管理用户身份登录 IAM Identity Center，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录 URL。要获取使用 IAM Identity Center 用户登录方面的帮助，请参阅《Amazon 登录用户指南》中的[登录 Amazon 访问门户](#)。

## 步骤 3：设置 IAM 权限

接下来，您必须创建一个 AWS Identity and Access Management (IAM) 策略，为用户提供一组基本权限（例如，创建 Amazon IVS 频道、获取直播信息和 auto-record-to-S3），并将该策略分配给用户。您可以在创建[新用户](#)时分配权限，也可以向[现有用户](#)添加权限。这两项程序如下。

有关更多信息（例如，了解 IAM 用户和策略、如何将策略附加到用户以及如何限制用户可以使用 Amazon IVS 执行的操作），请参见：

- [IAM 用户指南](#)中的创建 IAM 用户
- [Amazon IVS 安全性](#)中关于 IAM 和“IVS 的托管式策略”的信息。
- 对于“录制到 S3”功能：[使用服务相关角色](#)和 Amazon IVS 用户指南中的 [自动录制到 Amazon S3](#)。

您可以对 Amazon IVS 使用现有的 AWS 托管式策略，也可以创建新策略，该策略可自定义想要授予一组用户、组或角色的权限。下面介绍了这两种方法。

### 使用 IVS 权限的现有策略

在大多数情况下，您需要对 Amazon IVS 使用 AWS 托管式策略。IVS 安全性的 [IVS 的托管式策略](#)部分对其进行了全面描述。

- 使用 IVSReadOnlyAccess AWS 托管式策略，您的应用程序开发人员可以访问所有 IVS Get 和 List API 端点（低延迟和实时直播均适用）。
- 使用 IVSFullAccess AWS 托管式策略，您的应用程序开发人员可以访问所有 IVS API 端点（低延迟和实时直播均适用）。

### 可选：为 Amazon IVS 权限创建自定义策略

按照以下步骤进行操作：

1. 登录 Amazon 管理控制台，并通过以下网址打开 IAM 控制台：<https://console.aws.amazon.com/iam/>。
2. 在导航窗格中，选择策略，然后选择创建策略。这时将会打开指定权限窗口。
3. 在指定权限窗口中，选择 JSON 选项卡，然后复制下列 IVS 策略并粘贴到策略编辑器文本区域。[该策略不包括所有 Amazon IVS 操作。您可以根据需要添加/删除（允许/拒绝）端点访问权限。有关 IVS 端点的详细信息，请参阅 [IVS Low-Latency Streaming API Reference](#)。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ivs:CreateChannel",
        "ivs:CreateRecordingConfiguration",
        "ivs:GetChannel",
        "ivs:GetRecordingConfiguration",
        "ivs:GetStream",
        "ivs:GetStreamKey",
        "ivs:GetStreamSession",
        "ivs:ListChannels",
        "ivs:ListRecordingConfigurations",
        "ivs:ListStreamKeys",
        "ivs:ListStreams",
        "ivs:ListStreamSessions"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:DescribeAlarms",
        "cloudwatch:GetMetricData",
        "s3:CreateBucket",
        "s3:GetBucketLocation",
        "s3:ListAllMyBuckets",
        "servicequotas:ListAWSDefaultServiceQuotas",
        "servicequotas:ListRequestedServiceQuotaChangeHistoryByQuota",
        "servicequotas:ListServiceQuotas",
        "servicequotas:ListServices",
        "servicequotas:ListTagsForResource"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:AttachRolePolicy",
        "iam:CreateServiceLinkedRole",
        "iam:PutRolePolicy"
      ]
    }
  ]
}

```

```
    ],  
    "Resource":  
    "arn:aws:iam::*:role/aws-service-role/ivs.amazonaws.com/  
    AWSServiceRoleForIVSRecordToS3*"  
  }  
]  
}
```

4. 继续在指定权限窗口中，选择下一步（滚动到窗口底部即可看到此按钮）。这时将打开检查并创建窗口。
5. 在检查并创建窗口中，输入策略名称，此外还可以选择添加描述。记下策略名称，因为您在创建用户时需要使用该名称（如下文所述）。选择 Create policy（创建策略）（位于窗口底部）。
6. 您将返回到 IAM 控制台窗口，您应该会在该窗口中看到一条横幅，确认您的新策略已创建。

## 创建新用户并添加权限

### IAM 用户访问密钥

IAM 访问密钥包含一个访问密钥 ID 和一个秘密访问密钥。它们用于对您向 Amazon 发出的编程请求进行签名。如果没有访问密钥，您可以从 Amazon 管理控制台创建。作为最佳实践，请勿创建根用户访问密钥。

仅当创建访问密钥时，您才能查看或下载秘密访问密钥。以后您无法恢复它们。但是，您随时可以创建新的访问密钥；您必须拥有执行所需 IAM 操作的权限。

请务必安全地存储访问密钥。切勿与第三方共享（即使查询似乎来自 Amazon）。有关更多信息，请参阅《IAM 用户指南》中的[管理 IAM 用户的访问密钥](#)。

### 过程

按照以下步骤进行操作：

1. 在导航窗格中，选择用户，然后选择创建用户。这时将会打开指定用户详细信息窗口。
2. 在指定用户详细信息窗口中：
  - a. 在用户详细信息下，键入要创建的新用户名。
  - b. 选中授予用户访问亚马逊云科技管理控制台的权限。
  - c. 出现提示时，请选择我要创建 IAM 用户。
  - d. 在控制台密码下，选择自动生成的密码。

- e. 选中用户下次登录时必须修改密码旁的复选框。
  - f. 请选择 Next ( 下一步 )。这时将会打开设置权限窗口。
3. 在设置权限下，选择直接附加策略。这时将会打开权限策略窗口。
  4. 在搜索框中，输入 IVS 策略名称 ( AWS 托管式策略或您之前创建的自定义策略 )。找到该策略后，选中复选框以选择该策略。
  5. 选择下一步 ( 位于窗口底部 )。这时将打开检查并创建窗口。
  6. 在检查并创建窗口中，确认所有用户详细信息均正确无误，然后选择创建用户 ( 位于窗口底部 )。
  7. 这时将会打开找回密码窗口，其中包含您的控制台登录详细信息。妥善保存好此信息，以备将来参考。完成后，选择返回用户列表。

## 向现有用户添加权限

按照以下步骤进行操作：

1. 登录 Amazon 管理控制台，并通过以下网址打开 IAM 控制台：<https://console.aws.amazon.com/iam/>。
2. 在导航窗格中，选择用户，然后选择要更新的现有用户名。( 单击选择用户名；不要选中选择框。 )
3. 在摘要页面的权限选项卡中，选择添加权限。这时将会打开添加权限窗口。
4. 选择 Attach existing policies directly ( 直接附加现有策略 )。这时将会打开权限策略窗口。
5. 在搜索框中，输入 IVS 策略名称 ( AWS 托管式策略或您之前创建的自定义策略 )。找到该策略后，选中复选框以选择该策略。
6. 选择下一步 ( 位于窗口底部 )。这时将会打开检查窗口。
7. 在检查窗口中，选择添加权限 ( 位于窗口底部 )。
8. 在 Summary ( 摘要 ) 页面上，确认已添加 IVS 策略。

## 步骤 4：使用可选录制创建通道

Amazon IVS 通道存储与实时流相关的配置信息。您首先创建一个通道，然后使用通道的流密钥向其提供视频以启动实时流。

作为通道创建的一部分，将分配以下项目：

- 提取服务器识别接收流的特定 Amazon IVS 组件以及提取协议 ( RTMPS 或 RTMP )。

- 当您创建通道并使用通道授权流传输时，Amazon IVS 分配流密钥。将流密钥视为秘密，因为它允许任何人流传输到通道。
- 播放 URL 标识要开始播放特定通道的终端节点。此终端节点可以在全球范围内使用。它会自动从 Amazon IVS 全球内容分发网络中选择最佳位置，以供查看者流传输视频。（请注意，Amazon IVS 不支持用于播放的自定义域。请勿使用您自己的域代理播放 URL；这不起作用，并会导致出现问题。）

您可以通过 Amazon IVS 控制台或 Amazon CLI 创建通道，无论是否录制。将在下面讨论通道创建和录制。

## 自动录制到 Amazon S3

您可以选择启用通道录制。如果启用了自动录制到 S3 功能，则通道上的所有流都会记录并保存到您拥有的 Amazon S3 存储桶中。随后，录制可用于按需播放。

该设置是一个高级选项。默认情况下，在创建通道时禁用录制。

在您设置通道进行录制之前，必须创建录制配置。这是一个资源，它指定存储通道录制流的 Amazon S3 位置。您可以使用控制台或 CLI 来创建和管理录制配置；这两个过程如下所示。创建录制配置后，您可以在创建通道时（如下所述）或之后，通过更新现有通道将其与通道关联。（在 API 中，请参阅 [CreateChannel](#) 和 [UpdateChannel](#)。）您可以将多个通道与相同的录制配置相关联。您可以删除不再与任何通道关联的录制配置。

请注意以下限制：

- 您必须拥有 S3 存储桶。也就是说，设置要录制通道的账户必须拥有存储录制的 S3 存储桶。
- 通道、录制配置和 S3 位置必须位于同一亚马逊云科技区域。如果您在其他区域创建通道并想要录制它们，则还必须在这些区域中设置录制配置和 S3 存储桶。

录制到您的 S3 存储桶需要使用您的亚马逊云科技凭证进行授权。要向 IVS 提供所需的访问权限，在创建录制配置时自动创建 Amazon IAM [服务相关角色](#) (SLR)：SLR 仅限于针对特定存储桶为 IVS 提供写入权限。

若要在启用录制后禁用此功能，请参阅 [the section called “如何禁用录制”](#)。

请注意，流传输位置与 Amazon 之间或 Amazon 内部的网络问题可能会在录制流时导致一些数据丢失。在这些情况下，Amazon IVS 将实时流优先于录制。为了实现冗余，请通过流传输工具在本地录制。

有关详细信息（包括如何在录制的文件上设置后期处理或 VOD 播放），请参阅[自动录制到 Amazon S3](#)。

## 控制台说明

这些步骤分为三个阶段：初始通道设置、设置为自动录制到 Amazon S3（可选）和最终通道创建。

### 初始化通道设置

1. 打开 [Amazon IVS 控制台](#)。


（您还可通过 [Amazon 管理控制台](#) 访问 Amazon IVS 控制台。）

2. 从导航栏中，使用 Select a Region (选择区域) 下拉菜单选择一个区域。您的新通道将在此区域创建。
3. 在 Get started (开始使用) 框（右上角）中，选择 Create Channel (创建通道)。
4. 在 Channel configuration (通道配置) 下，接受 Default configuration (默认配置)。（可选）指定一个 Channel name (通道名称)。通道名称不是唯一的，但它们为您提供了一种方法来区分通道而不是通道 ARN (Amazon 资源名称)。

注意：自定义配置可用于指定某些非默认值，例如通道类型或 RTMP（而不是 RTMPS）提取。此处未记录自定义规格。



# Create channel [Info](#)

A channel is a unique configuration for streams. It includes broadcast configuration details (a server URL and stream key) for streaming software/hardware, and a playback URL for playing the stream. Channel configuration may affect pricing. [Amazon IVS Pricing](#) 

## ► How Amazon Interactive Video Service works

### Setup

#### Channel name

Maximum length: 128 characters. May include numbers, letters, underscores (\_) and hyphens (-).

#### Channel configuration

**Default configuration**  
Use the default video latency and configuration, optimized for live interactions.

**Custom configuration**  
Specify your own channel type and video latency configuration.

#### Channel type [Info](#)

Standard (broadcast and deliver live video up to 1080p Full HD, with transcoding and 1080p pass-through)

#### Video latency [Info](#)

Low (best for low-latency interactions with viewers)

#### Playback authorization [Info](#)

Disabled

#### Insecure ingest [Info](#)

Disabled

### Restrict playback [Info](#)

#### Playback restriction policy [Info](#)

Restrict playback by country and origin.

Enable playback restriction

5. 如果您想要自动录制到 Amazon S3，请继续以下的 [设置为自动录制到 Amazon S3 \( 可选 \)](#)。否则，跳过此步骤并直接前往 [最终通道创建](#)。

## 设置为自动录制到 Amazon S3 ( 可选 )

请按照以下步骤在创建新通道时启用录制：


1. 在创建通道页面的录制和存储流中，打开启用自动录制。显示其他字段，以选择现有 Recording configuration (录制配置) 或新建一个。

### Record and store streams [Info](#)

**Auto-record to S3** [Info](#)  
For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

Recording configuration

 **Associated costs**  
There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

► **Tags** [Info](#)  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

2. 选择 Create recording configuration (创建录制配置)。此时会打开一个新窗口，其中选项用于创建 Amazon S3 桶并将其附加到新的录制配置。

## Create recording configuration ✕

### Recording configuration name – *optional*

*recording-configuration-1*

Maximum length: 128 characters. May include numbers, letters, underscores ( `_` ) and hyphens ( `-` ).

### Recording configuration

- Default configuration**  
Use the default settings for auto-recording video and thumbnails.

- Custom configuration**  
Specify your own video and thumbnail recording options.

### Recorded renditions

All renditions

### Thumbnail recording

Record at an interval - 60s

### Target thumbnail resolution

Source (same resolution as input stream)

### Thumbnail storage

Store thumbnails sequentially

### Merge fragmented streams

Disabled

### Storage

- Create a new Amazon S3 bucket
- Select an existing Amazon S3 bucket

### Bucket name

*ivs-stream-archive*

The bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#).

### 3. 填写以下字段：

- a. ( 可选 ) 输入 Recording configuration name ( 录制配置名称 )。
  - b. 在通道配置下，接受默认配置。注意：自定义配置可用于指定某些非默认值，例如录制的渲染或合并片段化的流。此处未记录自定义规格。
  - c. 输入 Bucket Name (存储桶名称)。
4. 选择 Create recording configuration ( 创建记录配置 )，以创建具有唯一 ARN 的新录制配置资源。创建录制配置通常在数秒钟内完成，但最多可能需要 20 秒钟。录制配置创建完成后，您将返回到 Create channel (创建通道) 窗口。这时，录制和存储流区域将显示您的新录制配置，其状态为活跃，此外还会显示您创建的 S3 存储桶 ( 存储 )。

## Record and store streams [Info](#)

**Auto-record to S3** [Info](#)

For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

**Recording configuration**

configuration-1
▼
↻
Create recording configuration

<p><b>State</b></p> <p><span style="color: green;">✔</span> Active</p>	<p><b>Storage</b></p> <p><a href="#">s3-bucket-name</a></p>	<p><b>Recording prefix</b> <a href="#">Info</a></p> <p>s3://ivs-r2s3-ivsstoragebucket-1kem14abgbit8/ivs/v1/298083573632/&lt;attached_channel_id&gt;/</p>
<p><b>Recorded renditions</b></p> <p>All renditions</p>	<p><b>Merge fragmented streams</b></p> <p>Disabled</p>	<p><b>Thumbnail recording</b></p> <p>At 60-second intervals</p>
<p><b>Thumbnail storage</b></p> <p>Store thumbnails sequentially</p>	<p><b>Thumbnail resolution</b></p> <p>Source (same resolution as input stream)</p>	

**Associated costs**

There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

**Tags** [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel
Create channel

## 最终通道创建

1. 选择 Create channel (创建通道)，使用唯一 ARN 创建新通道。此时将显示新通道的通道详情页面；请保持打开状态。（注意：如果您没有启用录制，Auto-record to S3 (自动记录至 S3) 将设置为 Disabled (已禁用)，并且屏幕上没有 Recording configuration (录制配置) 部分。）

# channel-live Info

Edit Delete

## General configuration

Channel name channel-live	Channel type Standard	Video latency Ultra-low
Playback authorization Disabled	Auto-record to S3 Enabled	ARN arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh

## ▶ Live stream

## Stream configuration Info

Reset stream key

Stream key [Show](#)  
 .....

Ingest server  
 rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/

## ▶ Other ingest options

## Playback configuration Info

Playback URL  
 https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8

## Recording configuration Info

Manage

Recording configuration <a href="#">configuration-1</a>	Storage <a href="#">s3-bucket-name</a>	Recording prefix s3://qazwsxedcrfv-deleteme-1/ivs/v1/1234567891012/a1b2c3d4e5f6/
--	---	---

## Stream sessions Info

< 1 >

Stream ID	Start time	Duration
-----------	------------	----------

No past streams to display  
No past streams

## Tags Info

Manage tags

Search tags

## 2. 重要提示：

- 在 Stream configuration (流配置) 区域，请注意 Ingest server (提取服务器) 和 Stream key (流密钥)。您将在下一步中使用它们，以设置流传输。
- 在 Playback configuration (播放配置) 区域，请注意 Playback URL (播放 URL)。稍后，您将使用它来回放您的流。

## CLI 说明

使用 Amazon CLI 创建通道是一个高级选项，需要先在计算机上下载并配置 CLI。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

执行以下两个步骤之一，具体取决于您是否在启用录制的情况下创建通道。

### 创建通道但不录制

1. 运行 `create-channel` 命令并传入一个可选名称：

```
aws ivs create-channel --name test-channel
```

2. 此时将返回一个新通道：

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "authorized": false,
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "latencyMode": "LOW",
    "name": "channel-live",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/abcdABCDefgh",
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "recordingConfigurationArn": "none",
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "tags": {}
  }
}
```

```

    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef"
  }
}

```

3. **重要提示：**记下 `ingestEndpoint`、`streamKey value` 和 `playbackUrl`。您将使用它们来设置流传输和播放。

## 使用录制创建通道

**先决条件：**在开始此过程之前，请创建一个 Amazon S3 存储桶，并记下其 ARN。请参阅 [Amazon S3 入门](#)。S3 存储桶必须位于您要创建录制配置的区域；请参阅以下第 1 步中的已知问题。

然后执行以下步骤以创建通道：

1. 运行 `create-recording-configuration` 命令并传入现有 Amazon S3 存储桶的 ARN：

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=test-bucket}
```

您还可以传递 `thumbnail-configuration` 参数以手动设置缩略图录制模式和缩略图间隔：

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=s3_bucket_name} --thumbnail-configuration recordingMode="INTERVAL",targetIntervalSeconds=60
```

( 可选 ) 传递 `recording-reconnect-window-seconds` 参数以启用片段化的流功能：

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=test-bucket} --recording-reconnect-window-seconds 60
```

**已知问题：**在 `us-east-1` 区域中，如果您使用 Amazon CLI 创建录制配置，则即使 S3 存储桶位于不同区域，也会返回成功。在本案例中，录制配置的 `state` 为 `CREATE_FAILED` ( 而不是 `ACTIVE` )。( 在其他区域中，如果存储桶位于不同区域，CLI 会正确返回失败结果。 )

**解决办法：**确保您的 S3 存储桶与录制配置位于同一区域。如果您在与 S3 存储桶不同的区域创建了录制配置，请删除该录制配置，并从正确的区域创建一个带有 S3 存储桶的新配置。

2. 这将返回带有唯一 ARN 的新录制配置。录制配置的状态为 `CREATING`，表示当前正在创建。

```
{
```



```

"recordingConfiguration": {
  "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
mhndauNa01te",
  "name": "configuration-1",
  "destinationConfiguration": {
    "s3": {
      "bucketName": "s3_bucket_name"
    }
  },
  "recordingReconnectWindowSeconds": 60,
  "state": "CREATING",
  "tags": {},
  "thumbnailConfiguration": {
    "recordingMode": "INTERVAL",
    "targetIntervalSeconds": 60
  }
}
}

```

3. 创建录制配置通常在数秒钟内完成，但最多可能需要 20 秒钟。要检查是否已创建录制配置，请运行 `get-recording-configuration` 命令：

```
aws ivs get-recording-configuration --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/mhndauNa01te"
```

4. 这将返回一个响应，指示录制配置已创建（state 是 ACTIVE）：

```

{
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
mhndauNa01te",
    "name": "configuration-1",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "s3_bucket_name"
      }
    },
    "recordingReconnectWindowSeconds": 60,
    "state": "ACTIVE",
    "tags": {},
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 60
    }
  }
}

```

```
    }  
  }  
}
```

5. 要创建通道并启用通道录制功能，请运行 `create-channel` 命令并传入录制配置 ARN：

```
aws ivs create-channel --name channel-live --recording-configuration-arn  
"arn:aws:ivs:us-west-2:123456789012:recording-configuration/mhndauNa01te"
```

- 或者，要在现有通道上启用录制功能，请运行 `update-channel` 命令并传入录制配置 ARN：

```
aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/  
abcdABCDefgh" --recording-configuration-arn "arn:aws:ivs:us-  
west-2:123456789012:recording-configuration/mhndauNa01te"
```

6. 这将为 `recordingConfigurationArn` 返回一个具有非“none”值的通道对象，表示已启用录制。  
(以下响应来自 `create-channel`。`update-channel` 响应不包括 `streamKey` 对象。)

```
{  
  "channel": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "authorized": false,  
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",  
    "insecureIngest": false,  
    "latencyMode": "LOW",  
    "name": "channel-live",  
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/  
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",  
    "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-  
configuration/mhndauNa01te",  
    "tags": {},  
    "type": "STANDARD"  
  },  
  "streamKey": {  
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",  
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",  
    "tags": {},  
    "value": "sk_us-west-2_abcdABCDefgh_567890abcdef"  
  }  
}
```

7. 重要提示：记下 `ingestEndpoint`、`streamKey` value 和 `playbackUrl`。您将使用它们来设置流传输和播放。

## 步骤 5：设置流式传输软件

您可以通过以下方式（低延迟）流式传输到 Amazon IVS：

- 支持 RTMPS 的原生 [IVS 广播 SDK](#)。我们推荐采用此方法，尤其是在生产场景中。
- [Amazon IVS 控制台](#) — 适用于测试流。
- 其他流式传输软件 — 下面描述了两个示例，即 [开放广播软件 \(OBS\)](#) 和 [FFmpeg](#)。它们都像 Amazon IVS 一样支持 RTMPS 和 RTMP。RTMPS 通过使用已加密 TLS 流实现高安全性。您可以使用任何支持 RTMPS 或 RTMP 的流式传输软件。

键编码器设置为关键帧间隔（2 秒）和分辨率/比特率/帧速率（相互关联）。有关编码器设置的更多信息，请参阅：

- Amazon IVS 用户指南中的 [流传输配置](#)。
- 此博客文章：[使用 Amazon Interactive Video Service 设置流传输](#)。

备注：

- Amazon IVS 流的最长持续时间为 48 小时。之后，流将终止，流传输会话将断开连接。成功重新连接（自动或手动）将启动新流。
- 如果您的编码器停止发送数据（例如，出现临时网络问题），Amazon IVS 将等待 30 秒钟。如果在此期间未收到直播数据，Amazon IVS 将断开连接。

## 使用 Amazon IVS 广播开发工具包进行流传输

要从 iOS 或 Android 应用程序中进行广播，您可以使用 Amazon IVS 广播开发工具包。广播开发工具包利用 Amazon IVS 架构，并将实现 Amazon IVS 的持续改进和新功能。作为本机移动广播开发工具包，它旨在最大限度地减少对应用程序以及用户有权访问应用程序所在设备的性能影响。

要从 ... 广播	您可以使用 ...	注意事项
您的 Android 或 iOS 应用程序	Amazon IVS Android 或 iOS 广播 SDK	作为本机移动广播开发工具包，它旨在最大限度地减少对应用程序以及用户有权访问应用程序所在设备的性能影响。
一个 Web 环境	Amazon IVS Web 广播 SDK	作为一种 Web 广播 SDK，Amazon IVS Web 广播 SDK 让您能够使用 WebRTC 从 Web 环境进行广播。它提供浏览器和跨平台支持。

有关更多信息，请参阅 [IVS 广播 SDK](#)。

## 使用 Amazon IVS 控制台进行流式传输

### 1. 打开 [Amazon IVS 控制台](#)。

( 您还可通过 [Amazon 管理控制台](#) 访问 Amazon IVS 控制台。 )

2. 在导航窗格中，选择通道。( 如果导航窗格已折叠，请选择汉堡图标以将其展开。 )
3. 选择您要广播的通道，以转至该通道的详细信息页面。
4. 选择广播选项卡。( 这些选项卡在常规配置部分下。 )
5. 系统将提示您授予 IVS 控制台访问您摄像头和麦克风的权限；允许这些权限。
6. 在广播选项卡的底部，使用下拉框为麦克风和摄像头选择输入设备。
7. 要开始流式传输，请选择开始广播。
8. 要查看实时流，请转至播放选项卡。

注意：启动流后，预计会有短暂的延迟（通常不到 30 秒），然后才能在“播放”选项卡中查看。

您可以使用此功能同时向多个通道广播。

注意：从控制台进行流式传输会消耗资源，并且您将产生实时视频输入费用。要了解更多信息，请参阅 IVS 定价页面上的 [实时视频输入费用](#)。

## 通过 OBS Studio 进行流传输

([OBS Studio](#)) 是一个免费的开源软件套件，用于录制和实时流传输。OBS 提供实时源和设备捕获、场景合成、编码、录制和流传输。

按照以下步骤让 OBS Studio 快速设置并运行。

1. 下载并安装此软件：<https://obsproject.com/download>。
2. 运行 OBS Studio 自动配置向导，其在首次加载 OBS Studio 时会显示。执行以下步骤并接受默认值。
3. 在 Stream Information (流信息) 中，从 Service (服务) 下拉列表中选择 Custom... (自定义.....)，然后输入 Server (Ingest server) [服务器 (提取服务器)] 和 Stream Key (流密钥)。

如果您使用 Amazon IVS 控制台创建了通道：

- 您在 OBS 中输入的 Server (服务器) 是控制台中的 Ingest server (提取服务器)：

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/
```

- 您在 OBS 中输入的 Stream key (流密钥) 是控制台中的 Stream key (流密钥)：

```
sk_us-west-2_abcd1234efgh5678ijkl
```

如果您使用 Amazon CLI 创建了通道：

- 您在 OBS 中输入的 Server (服务器) 是以下四个选项的组合：

- 提取协议：`rtmps://` (注意：如果您为不安全的提取流式传输创建了通道，请指定 `rtmp://`)

- CLI 响应中的提取终端节点：

```
a1b2c3d4e5f6.global-contribute.live-video.net
```

- 端口：443
- 路径：`/app/`

完整条目为：

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/
```

- 您在 OBS 中输入的 Stream key (流密钥) 是 CLI 响应的 streamKey value (流密钥值)：

```
sk_us-west-2_abcd1234efgh5678ijkl
```

4. 有关视频输出分辨率和比特率的信息，请参阅《Amazon IVS 流传输配置》中的[通道类型](#)。如果 OBS 向导选择的任一值超过了 Amazon IVS 允许的值，则应手动调整这些值以避免与 Amazon IVS 连接失败。向导完成后：

- a. 要调整视频分辨率，请使用 Settings > Video > Output (Scaled) Resolution [设置 > 视频 > 输出 (缩放) 分辨率]。

- b. 要调整视频比特率，请使用 Settings > Output > Streaming > Video Bitrate ( 设置 > 输出 > 流传输 > 视频比特率 )。
5. 我们建议使用 2 秒的 Keyframe Interval ( 关键帧间隔 )，以提高流稳定性并避免查看者播放时缓冲。向导完成后，转到 Settings > Output > Output Mode ( 设置 > 输出 > 输出模式 )，选择 Advanced ( 高级 )，然后在 Streaming ( 流传输 ) 选项卡上，确保 Keyframe Interval ( 关键帧间隔 ) 为 2。
6. 在 OBS Studio 主窗口中，选择 Start Streaming ( 开始流传输 )。

有关使用 OBS Studio 进行流传输的更多信息，请参阅 [OBS Studio 快速入门](#)。

稍后，您可以手动修改 OBS 设置：

1. 选择 Settings (设置) > Stream (流)。
2. 从下拉列表中选择 Custom (自定义)。
3. 粘贴至 Server (服务器) 和/或 Stream Key (流密钥)。

您可以随时再次运行向导：选择 Tools (工具) > Auto-Configuration Wizard (自动配置向导)。

( 可选 ) 在 Settings (设置) > General (常规) 中，启用本地录制以保存实时流以供日后使用。如前所述，广播与亚马逊云科技之间或亚马逊云科技内部出现网络问题可能会在录制流时导致一些数据丢失。在这些情况下，Amazon IVS 将实时流优先于录制。通过流传输工具进行本地录制可提供冗余。

建议定期检查 OBS Studio 更新，并更新到最新版本。( 例如，如果出现“无法连接到服务器”错误，则可能是使用了不支持 RTMPS 的旧版 OBS Studio。 )

## 使用 FFmpeg 对录制的视频进行流传输

按照以下步骤进行操作：

1. 下载并安装 FFmpeg：<https://www.ffmpeg.org/download.html>。
2. 将 \$VIDEO\_FILEPATH 设置为要进行流传输的 MP4 视频的位置：

```
VIDEO_FILEPATH=/home/test/my_video.mp4
```

3. 设置 STREAM\_KEY 为你的 StreamKey 值：

```
STREAM_KEY=sk_us-west-2_abcd1234efgh5678ijkl
```

4. 将 `INGEST_ENDPOINT` 设置为您的 `ingestEndpoint` ( Amazon CLI 中 ) :

```
INGEST_ENDPOINT=a1b2c3d4e5f6.global-contribute.live-video.net
```

5. 使用以下终端命令开始流传输 ( 显示在一行中 ) :

```
ffmpeg -re -stream_loop -1 -i $VIDEO_FILEPATH -r 30 -c:v libx264 -pix_fmt yuv420p  
-profile:v main -preset veryfast -x264opts "nal-hrd=cbr:no-scenecut" -minrate  
3000 -maxrate 3000 -g 60 -c:a aac -b:a 160k -ac 2 -ar 44100 -f flv rtmps://  
$INGEST_ENDPOINT:443/app/$STREAM_KEY
```

请注意，以上命令是一个示例。对于生产流传输，请根据您的需求调整参数。

## 步骤 6：查看您的实时流

要查看您的实时流：

1. 打开 [Amazon IVS 控制台](#)。

( 您还可通过 [Amazon 管理控制台](#) 访问 Amazon IVS 控制台。 )

2. 在导航窗格中选择 Live channels ( 直播通道 )。( 如果导航窗格已折叠，请首先选择汉堡包图标以将其打开。 )
3. 选择要查看其流的通道，以转到该通道的详细信息页面。

此实时流正在该页面的 Live stream (实时流) 部分播放。

**注意：**从控制台播放会消耗资源，并且会产生实时视频输出费用。要了解更多信息，请参阅 IVS 定价页面上的[实时视频输出费用](#)。

**注意：**开始流传输后，在控制台中查看流之前会有短暂的延迟 ( 最长 30 秒，通常更短 )。

## 步骤 7：查看您的服务限额限制 ( 可选 )

所有账户都对并发查看者数量和并发广播数量有限制。确保您的限制足够，并在需要时请求增加限额，尤其是在您计划大型流传输活动时。有关详细信息，请参阅 [服务限额 \( 低延迟流式传输 \)](#)。

## 第 8 步：阻止不想要的内容和浏览者（推荐）

恶意用户可能会尝试在您的平台上重新播放不良内容（例如职业体育节目），或者未经许可尝试将您平台的直播嵌入其他网站。这种流式传输可以显著增加您的应用程序提供的直播视频数量以及相关成本，而不会增加您的业务价值。除了为您提供停止活跃直播的控制措施外，Amazon IVS 还提供资源来帮助您首先检测和防止此类行为；请参阅[不想要的内容和观看者](#)。

要将播放限制在特定的来源和/或国家，请使用播放限制政策。

### 控制台说明（播放限制政策）

#### 1. 创建播放限制政策

- a. [打开 Amazon IVS 控制台](#)。在左侧导航窗格中，选择播放安全 > 播放限制策略。
- b. 选择创建策略。
- c. （可选）为策略命名。
- d. （可选）切换严格原产地强制执行（参见下面的注释）。
- e. 指定允许的国家/地区和允许的起运地。
- f. 选择创建策略。

#### 2. 将此政策附加到新的或现有的频道

- a. 创建新频道或编辑现有频道。
- b. 在“限制播放”部分（“创建频道”或“更新频道”窗口的）中，选择“启用播放限制”。
- c. 从播放限制策略下拉列表中，选择您在步骤 1 中创建的策略。
- d. 选择“创建频道”（用于新频道）或“保存”（更新现有频道）。

关于严格执行原产地的注意事项：这是一个可选设置，可用于加强使用允许的起点指定的起源限制。默认情况下，来源限制仅适用于多变体播放列表。如果启用了严格的来源强制执行，则服务器将强制要求请求来源与所有播放请求（包括多变体播放列表、变体播放列表和片段）的令牌相匹配。这意味着所有客户端（包括非浏览器客户端）都必须在每个请求中提供有效的源请求标头。使用 `setOrigin` 方法在 IVS iOS 和 Android 回放器 SDK 中设置标头。除 iOS Safari 之外，它会在 Web 浏览器中自动设置。对于 iOS Safari，您需要将 `crossorigin="anonymous"` 添加到视频元素，以确保发送源请求标头。示例：`<video crossorigin="anonymous"></video>`。

关于 IP 地址和国家/地区之间映射的注意事项：IVS 使用第三方数据库确定用户的位置。IP 地址和国家/地区之间映射的准确性因区域而异。根据最近的测试，整体准确性为 99.8%。如果 IVS 无法确定用户的位置，IVS 将提供用户请求的内容。



## CLI 说明 ( 播放限制政策 )

1. 创建播放限制政策。下面是一个例子。对于 *allowed-countries* 和 *allowed-origins* 字段，将以下示例值替换为实际值，或者根据您的用例删除一个或两个字段。

```
aws ivs create-playback-restriction-policy --name test-playback-restriction-policy
--enable-strict-origin-enforcement --allowed-countries "US","JP" --allowed-origins
"https://example1.com","https://*.example2.com"
```

这将返回新的播放限制政策。有关其字段，请参阅 [PlaybackRestrictionPolicy](#) 《IVS 低延迟流媒体 API 参考》。

2. 将新政策附加到频道。对于现有频道，运行 `update-channel` 并传入上一步中创建的播放限制策略的 ARN：

```
aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh" --playback-restriction-policy-arn "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/abcdABCDefgh"
```

对于新频道，请在 [创建频道时添加--playback-restriction-policy-arn](#) 声明。

## 如何禁用录制

要在现有通道上禁用 Amazon S3 录制，请执行以下操作：

- 控制台 – 在相关通道的详细信息页面的 Record and store (记录和存储) 部分，选择 Disabled (已禁用)，然后选择 Save Channel (保存通道)。这将删除录制配置与通道的关联；该通道上的流将不再被录制。
- CLI – 运行 `update-channel` 命令并将录制配置 ARN 作为空字符串传入：

```
aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh" --recording-configuration-arn ""
```

这将返回一个通道对象，其中包含 `recordingConfigurationArn` 的空字符串，表示录制已禁用。

# 在 Amazon IVS 流中启用多台主机

Amazon Interactive Video Service ( IVS ) 使开发人员能够构建应用程序，将来自多个广播者 ( 也称为主持人 ) 的视频合并为一个实时流。

使用案例包括：

- 嘉宾席位：广播者可以邀请观众加入广播。这为卡拉 OK 和问答等协作内容打开了大门。
- 对战 ( VS ) 模式：广播者相互配对以竞争 ( 例如，在歌唱比赛中 )。
- 群组广播：多个发言者可以在大量观众面前对谈。

要向实时流添加多个广播者，您需要同时使用 IVS 实时直播和 IVS 低延迟直播。IVS 实时直播用于合并视频和音频流；而低延迟直播则用于将合并的流传播给查看者。

实时直播提供了一种名为“舞台”的资源，这是一个广播者 ( 主持人 ) 可以实时交换音频和视频的虚拟空间。然后您可以将“舞台”广播到通道让更多的观众看到，并构建让观众“登上舞台”参与实时对话的应用程序。

有关 IVS 实时直播的更多信息，请参阅：

- [IVS Real-Time Streaming User Guide](#)
  - IVS 广播 SDK 包含实时功能。请参阅这些 SDK 的指南：[Web](#)、[Android](#) 和 [iOS](#)，尤其是“发布和订阅”部分。
- [IVS Real-Time Streaming API Reference](#)

## 开始使用

### 控制台说明

要创建新舞台并为该舞台创建参与者令牌，请按照下列步骤操作：

1. 打开 [Amazon IVS 控制台](#)。

( 您还可通过 [Amazon 管理控制台](#) 访问 Amazon IVS 控制台。 )

2. 请在左侧导航窗格中选择舞台，然后选择创建舞台。此时将显示创建舞台窗口。

Amazon IVS > Video > Stages > Create stage

## Create stage [Info](#)

A stage allows participants to send and receive video and audio with others in real time. You can broadcast a stage to a channel, allowing viewers to see and hear stage participants without needing to join the stage directly. [Learn more](#) [↗](#)

### ▶ How Amazon IVS stages work

### Setup

Stage name – *optional*

Maximum length: 128 characters. May include numbers, letters, underscores ( `_` ) and hyphens ( `-` ).

### ▶ Tags [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel

Create stage

3. ( 可选 ) 输入舞台名称。请选择创建舞台以创建舞台。此时将显示新舞台的舞台详细信息页面。
4. 请选择创建参与者令牌。
5. 在创建参与者令牌对话框中，输入用户 ID 并选择创建参与者令牌。参与者令牌表的顶部将显示令牌。单击“复制令牌”图标（参与者令牌的左侧）以复制令牌。

## CLI 说明

使用 Amazon CLI 是一个高级选项，需要先在计算机上下载并配置 CLI。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

现在，您可以使用 CLI 创建和管理资源。舞台 API 在 `ivs-realtime` 命名空间下。例如，要创建舞台，以执行以下操作：

```
aws ivs-realtime create-stage --name "test-stage"
```

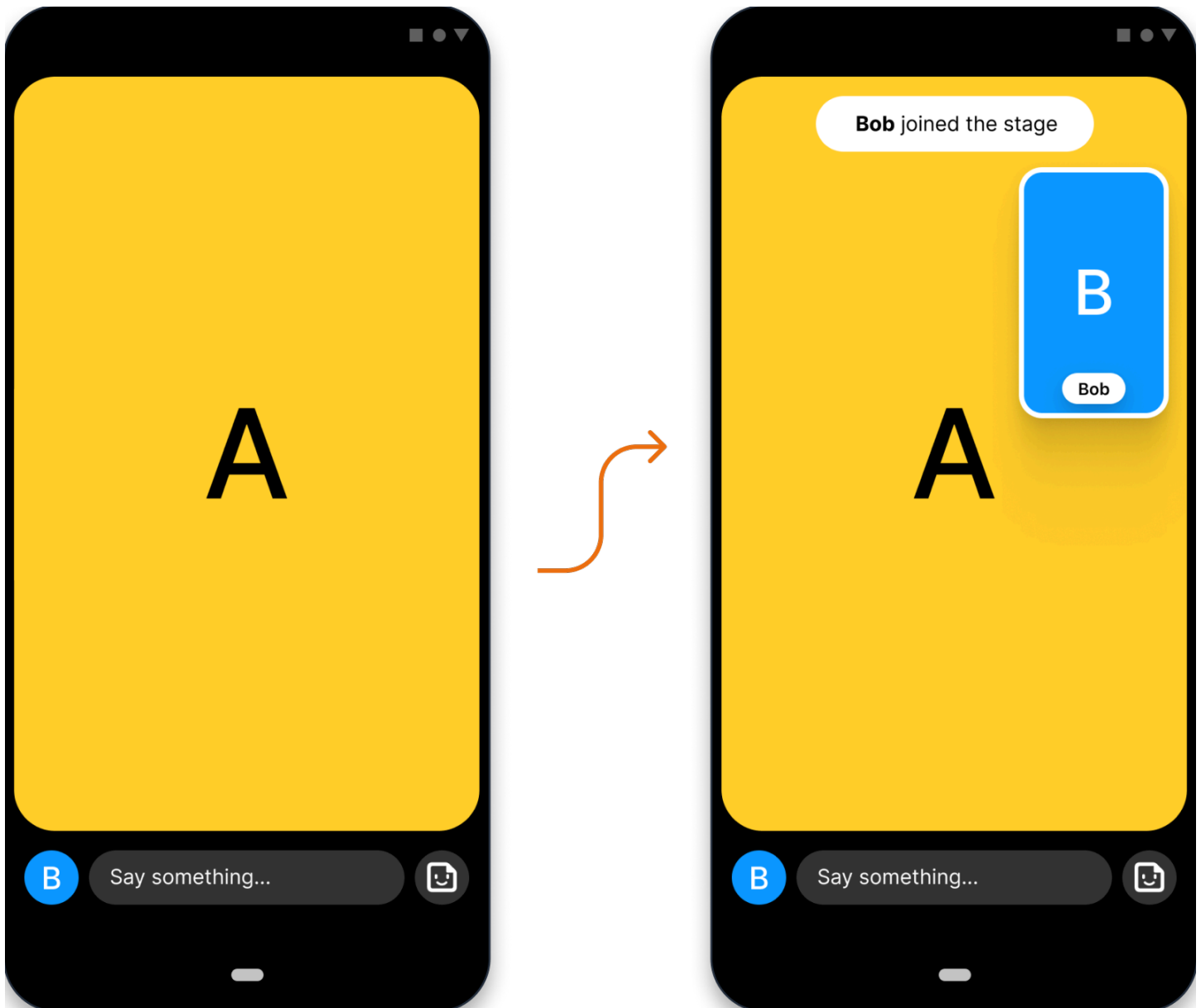


- 减少客户端负载 — 通过服务器端合成，组合音频和视频源的负担从单个客户端设备转移到服务器本身。服务器端合成使客户端设备无需使用其 CPU 和网络资源来合成视图并将其传输到 IVS。
- 弹性 — 通过将合成过程集中到服务器上，广播变得更加强大。即使发布者设备遇到技术限制或网络波动，服务器也可以进行调整，为所有受众提供更流畅的直播。
- 带宽效率 — 由于服务器负责合成，因此舞台发布者不必花费额外的带宽将视频广播到 IVS 频道。

有关更多信息，请参阅 IVS Real-Time User Guide 中的 [Server-Side Composition](#)。

## 演示

场景：Alice ( A ) 正在她的 Amazon IVS 通道上广播，想邀请 Bob ( B ) 作为嘉宾登上舞台。（在真实的广播中，A 和 B 将是 Alice 和 Bob 的照片。）



## 1. 创建舞台

以下是使用 Amazon IVS 舞台 API 的 [CreateStage](#) 请求：

```
POST /CreateStage HTTP/1.1
Content-type: application/json
{
  "name": "string",
  "participantTokenConfigurations": [
    {
      "userId": "9529828585",
      "attributes": {"displayName": "Alice"}
```

```
    },
    {
      "userId": "4875935192",
      "attributes": {"displayName": "Bob"}
    }
  ]
}
```

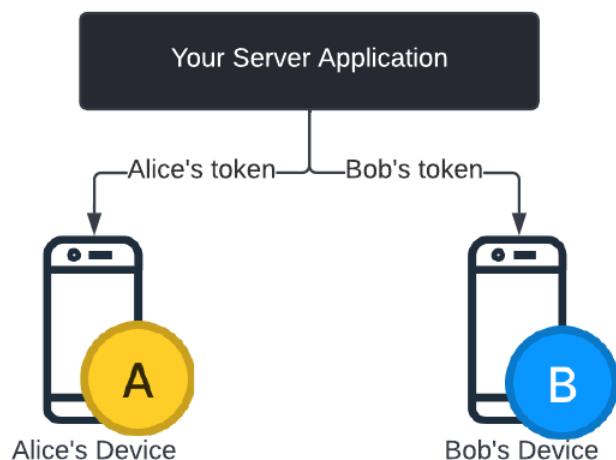
您可以在创建舞台时预先创建参与者令牌，如下所示。也可以通过调用 [CreateParticipantToken](#) 为现有舞台创建令牌。对于每个参与者，您可以传递自定义 `userId` 和一组 `attributes`。（重要提示：向所有舞台参与者公开 `attributes` 和 `userId` 请求字段。这些信息不应用于个人身份识别、机密或敏感信息。）

以下是对上述请求的网络响应：

```
HTTP/1.1 200
Content-type: application/json
{
  "stage": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
    "name": "alice-stage"
  },
  "participantTokens": [
    {
      "participantId": "e94e506e-f7...",
      "token": "eyJhbGciOiJ...",
      "userId": "9529828585",
      "attributes": {"displayName": "Alice"},
      "expirationTime": number
    },
    {
      "participantId": "b5c6a79a-6e...",
      "token": "eyJhbGciOiJ...",
      "userId": "4875935192",
      "attributes": {"displayName": "Bob"},
      "expirationTime": number
    }
  ]
}
```

## 2. 分发参与者令牌

客户端现在具备 Alice ( A ) 和 Bob ( B ) 令牌。默认情况下，令牌的有效期为 1 小时；您也可以选择在创建舞台时传递自定义 duration。令牌可用于加入舞台。



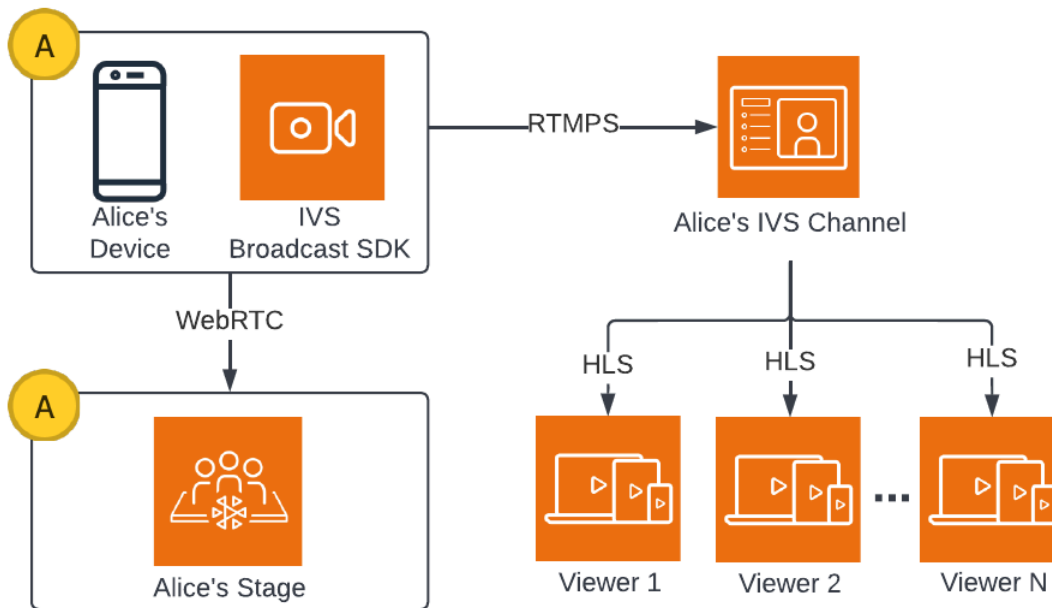
您将需要一种将令牌从服务器分发到每个客户端（例如，通过 WebSocket 通道）的方法。我们不提供此功能。

## 3. 加入舞台

参与者可以通过 Android 或 iOS 上的 Amazon IVS 广播 SDK 加入舞台。您可以配置每个参与者的视频质量。先看看 Alice 如何加入舞台。

以下是架构概述：





以下是加入舞台的 Android 代码示例。下面的代码片段将在 Alice 的设备上运行。在 `join()` 调用中，Alice 加入了舞台。上图显示了代码执行的结果：Alice 已加入舞台并正在向该舞台发布内容（此外还向她的频道广播，这是她在步骤 1 中开始进行的）。

```
// Create streams with the front camera and first microphone.
var deviceDiscovery = DeviceDiscovery(context)
var devices : List<Device> = deviceDiscovery.listLocalDevices()
var publishStreams = ArrayList<LocalStageStream>()

// Configure video quality if desired
var videoConfiguration = StageVideoConfiguration()

// Create front camera stream
var frontCamera = devices.find { it.descriptor.type ==
    Device.Descriptor.DeviceType.Camera && it.descriptor.position ==
    Device.Descriptor.Position.FRONT }
var cameraStream = ImageLocalStageStream(frontCamera, videoConfiguration)
publishStreams.add(cameraStream)

// Create first microphone stream
var microphone = devices.find { it.descriptor.type ==
    Device.Descriptor.DeviceType.Microphone }
var microphoneStream = AudioLocalStageStream(microphone)
publishStreams.add(microphoneStream)
```

```
// A basic Stage.Strategy implementation that indicates the user always wants to
// publish and subscribe to other participants.
// Provides the front camera and first microphone as publish streams.

override fun shouldPublishFromParticipant(stage: Stage, participantInfo:
    ParticipantInfo) : Boolean {
    return true
}

override fun shouldSubscribeToParticipant(stage: Stage, participantInfo:
    ParticipantInfo) : Stage.SubscribeType {
    return Stage.SubscribeType.AUDIO_VIDEO
}

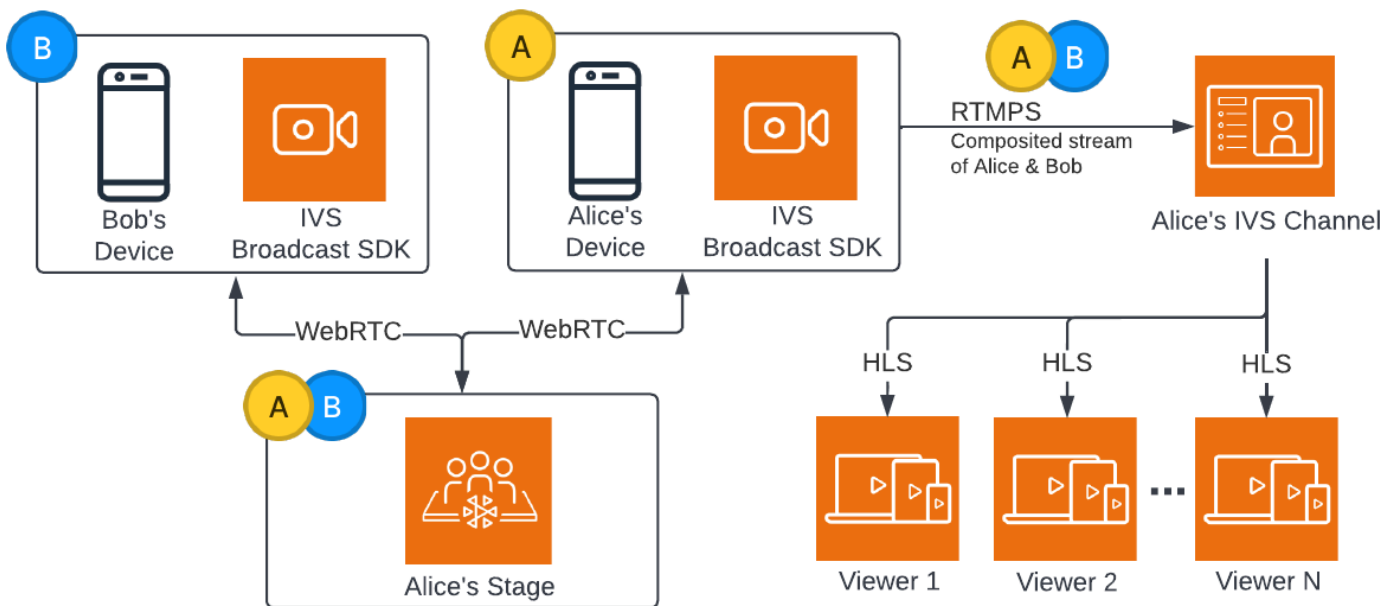
override fun stageStreamsToPublishForParticipant(stage: Stage, participantInfo:
    ParticipantInfo): List<LocalStageStream> {
    return publishStreams
}

// Create Stage using the strategy and join
var stage = Stage(context, token, strategy)

try {
    stage.join()
} catch (exception: BroadcastException) {
    // handle join exception
}
```

## 4. 广播舞台

### 客户端合成



以下是用于广播舞台的 Android 代码示例：

```
var broadcastSession = BroadcastSession(context, broadcastListener, configuration,
    null)

// StageRenderer interface method to be notified when remote streams are available
override fun onStreamsAdded(stage: Stage, participantInfo: ParticipantInfo, streams:
    List<StageStream>) {

    var id = participantInfo.participantId

    // Create mixer slot for remote participant
    var slot = BroadcastConfiguration.Mixer.Slot.with { s ->
        s.name = id
        // Set other properties as desired
        ...
        s
    }

    broadcastSession.mixer.addSlot(slot)

    // Attach remote stream devices, bind to mixer slot
```

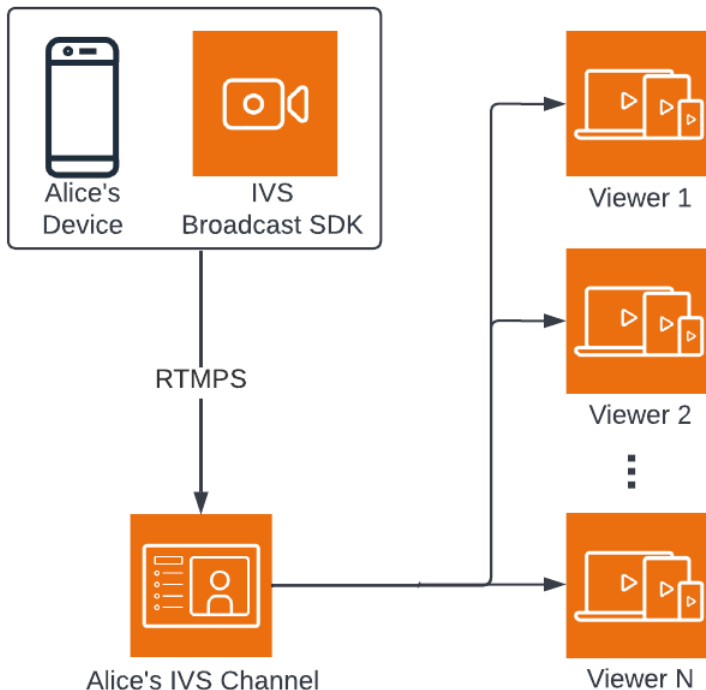
```
streams.forEach { stream ->
    broadcastSession.attachDevice(stream.getDevice())
    broadcastSession.mixer.bind(stream.getDevice(), id)
}

// Start broadcasting
try {
    broadcastSession.start(IVS_RTMP_URL, IVS_STREAM_KEY)
} catch (exception: BroadcastException) {
    // handle exception
}
```

Android 和 iOS Amazon IVS 广播 SDK 具有由参与者状态（例如 `onStreamsAdded` 和 `onStreamsRemoved`）触发的回调，以简化动态界面的构建。如代码示例的第一部分所示：当 Bob 的视频和音频可用时，Alice 会通过 `onStreamsAdded` 回调得到通知。

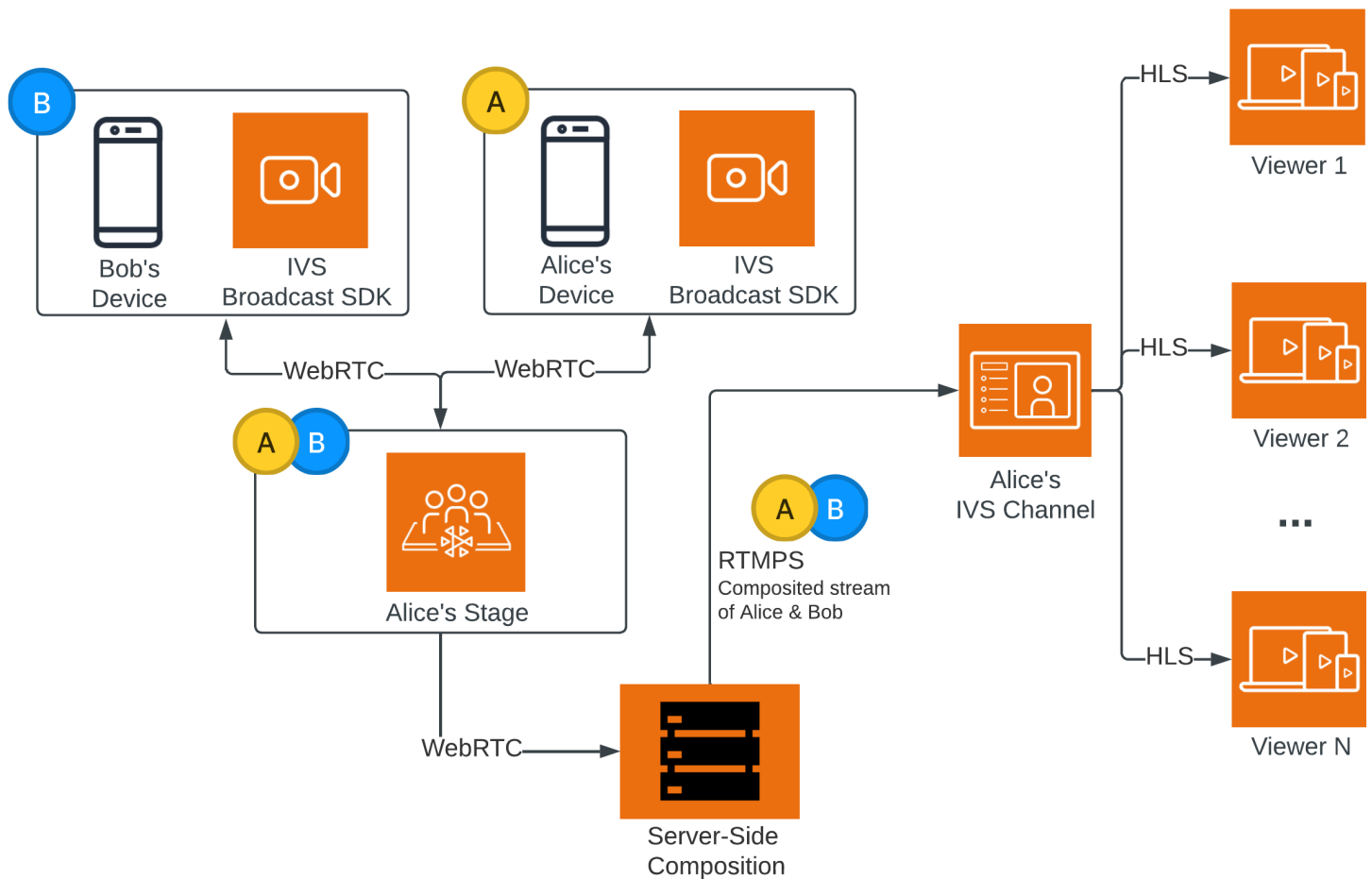
然后，Alice 可以将 Bob 的视频和音频添加到混合器中，将其包含在 RTMP 广播中，让通道的更多观众看到和听到。如代码示例的其余部分所示。

现在，Alice 正在通过 Amazon IVS Android 广播 SDK 向多位观众广播。以下是架构上的呈现：



## 服务器端合成

为了进行比较，这里提供了[服务器端合成](#)的工作原理。（有关详细信息，请参阅 [IVS Real-Time User Guide](#) 中的 [Server-Side Composition](#)。）



## 监控 Amazon IVS 低延迟直播功能

您可以使用 Amazon CloudWatch 监控 Amazon Interactive Video Service (IVS) 资源。CloudWatch 可从 Amazon IVS 收集和處理原始數據，並將數據處理為便於讀取的近乎實時的指標。這些統計數據會保存 15 個月，從而使您能夠從歷史角度了解您的 Web 应用程序或服務的執行情況。您可以設置用於特定閾值的警報，並在達到相應閾值時發送通知或執行操作。有關詳細信息，請參閱 [CloudWatch 用戶指南](#)。

指標上的時間戳表示累積指標數據的時間段開始時間。例如，假設您在 01:02:00 時得到的每分鐘 LiveDeliveredTime 指標總和為 300 秒。這意味著在 01:02:00 至 01:02:59 的 1 分鐘期間，查看者可以觀看 5 分鐘的視頻。

對於指定為高分辨率的指標，第一個數據點會在流開始數秒後顯示。我們建議您在發出指標請求時指定 5 秒的時間段。（請參閱 Amazon CloudWatch 用戶指南中的 [分辨率](#)。）對於其他指標，數據在其引用的時間戳的 1 分鐘內發出。

隨着時間的推移，高分辨率指標會彙總。隨着指標的老化，分辨率實際有所降低。以下是計劃：

- 1 秒的指標可用期為 3 小時。
- 60 秒的指標可用期為 15 天。
- 5 分鐘的指標可用期為 63 天。
- 1 小時的指標可用 455 天（15 個月）。

有關數據保留的最新信息，請在 [Amazon CloudWatch 常見問題](#) 中搜索“保留期”。

## 先決條件

- 您必須擁有具有充足 IAM 權限的亞馬遜云科技賬戶，才能與流運行狀況 API 和 CloudWatch 指標進行交互。有關具體步驟，請參閱 [Getting Started with IVS Low-Latency Streaming](#)。
- 您必須創建一個通道然後才能開啟流。相關信息請參閱 [IVS Low-Latency Streaming User Guide](#)：
  - 有關創建通道的說明，請參閱 Getting Started with IVS Low-Latency Streaming 中的 [Create a Channel](#)。
  - 有關啟用流的說明，請參閱 Getting Started with IVS Low-Latency Streaming 中的 [Set Up Streaming Software](#)。
  - 有關編碼器配置的詳細信息，請參閱 [Amazon IVS 流傳輸配置](#)。

## 访问流会话数据

使用 `listStreamSessions` 端点，您可以访问通道拥有的流列表，事件最长 60 天。此列表可能包括实时流会话（由空 `endTime` 表示）。

您可以通过 `getStreamSession` 端点获取特定流的会话数据。如果未指定 `streamId` 参数，端点返回最新的会话。此外，您可以定期调用终端节点以获取流的最新事件（最多最新的 500 个事件）。

### 控制台说明

1. 打开 [Amazon IVS 控制台](#)。

（您还可通过[亚马逊云科技管理控制台](#)访问 Amazon IVS 控制台。）

2. 在导航窗格中，选择 Channels（通道）。（如果导航窗格已折叠，请首先选择汉堡包图标以将其打开。）
3. 选择一个通道以转至其详细信息页面。
4. 向下滚动页面，直至看到 Stream sessions（流会话）部分。
5. 选择要访问的会话的流 ID，以查看其会话详细信息，包括 Amazon CloudWatch 高精度指标的图表。

或者，如果一个或多个通道已经运行：

1. 打开 [Amazon IVS 控制台](#)。
2. 在导航窗格中选择 Live channels（直播通道）。（如果导航窗格已折叠，请首先选择汉堡包图标以将其打开。）
3. 从列表中选择一个直播通道，在拆分视图中访问其会话详细信息。

### AWS SDK 说明

使用 Amazon SDK 访问流会话数据是一种高级选项，需要您先下载开发工具包并在应用程序上配置。以下是使用 JavaScript 的 Amazon SDK 的说明。

先决条件：要使用以下代码示例，您需要将 Amazon JavaScript SDK 加载到应用程序中。有关详细信息，请参阅 [Amazon SDK for JavaScript 入门](#)。

```
// This first call lists up to 50 stream sessions for a given channel.  
const AWS = require("aws-sdk");
```

```
const REGION = 'us-west-2';
let channelArn = USE_YOUR_CHANNEL_ARN_HERE;

AWS.config.getCredentials(function(err) {
  if (err) console.log(err.stack);
  // credentials not loaded
  else {
    console.log("Access key:", AWS.config.credentials.accessKeyId);
  }
});

AWS.config.update({region: REGION});
var ivs = new AWS.IVS();

// List Stream Sessions
async function listSessions(arn) {
  const result = await ivs.listStreamSessions({"channelArn": arn}).promise();
  console.log(result.streamSessions);
}
listSessions(channelArn);

// Get Stream Session
async function getSession(arn, id) {
  const result = await ivs.getStreamSession({"channelArn": arn, "streamId":
id}).promise();
  console.log(result);

  // This function polls every 3 seconds and prints the latest IVS stream events.
  setInterval(function(){
    console.log(result.streamSession.truncatedEvents);
  }, 3000);
}
getSession(channelArn);
```

## CLI 说明

使用 Amazon CLI 访问流会话数据是一种高级选项，需要您先下载 CLI 并在应用程序上配置。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

### 1. 列出流会话：

```
aws ivs list-stream-sessions --channel-arn <arn>
```



## 2. 使用流的 streamId 获取特定流的流会话数据：

```
aws ivs get-stream-session --channel-arn <arn> --stream-id <streamId>
```

以下为 get-stream-session 调用的示例响应：

```
{
  "streamSession": {
    "startTime": "2021-10-22T00:03:57+00:00",
    "streamId": "st-1FQzeLONMT9XTKI431eLSol",
    "truncatedEvents": [
      {
        "eventTime": "2021-10-22T00:09:30+00:00",
        "name": "Session Ended",
        "type": "IVS Stream State Change"
      },
      {
        "eventTime": "2021-10-22T00:09:30+00:00",
        "name": "Stream End",
        "type": "IVS Stream State Change"
      },
      {
        "eventTime": "2021-10-22T00:03:57+00:00",
        "name": "Stream Start",
        "type": "IVS Stream State Change"
      },
      {
        "eventTime": "2021-10-22T00:03:50+00:00",
        "name": "Session Created",
        "type": "IVS Stream State Change"
      }
    ],
    "endTime": "2021-10-22T00:09:31+00:00",
    "ingestConfiguration": {
      "audio": {
        "channels": 2,
        "codec": "mp4a.40.2",
        "sampleRate": 48000,
        "targetBitrate": 160000
      },
      "video": {
        "avcLevel": "4.0",
```

```
        "avcProfile": "Baseline",
        "codec": "avc1.42C028",
        "encoder": "obs-output module (libobs version 27.0.1)",
        "targetBitrate": 3500000,
        "targetFramerate": 30,
        "videoHeight": 1080,
        "videoWidth": 1920
    }
},
"channel": {
    "name": "",
    "ingestEndpoint": "3f234d592b38.global-contribute.live-video.net",
    "authorized": false,
    "latencyMode": "LOW",
    "recordingConfigurationArn": "",
    "type": "STANDARD",
    "playbackUrl": "https://3f234d592b38.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.991729659840.channel.dY7LsluQX1gV.m3u8",
    "arn": "arn:aws:ivs:us-west-2:991729659840:channel/dY7LsluQX1gV"
}
}
```

## 按运行状况筛选流

为了轻松找到哪些流遇到问题，你可以使用 `listStreams` 来按“运行状况”筛选流。

### 控制台说明

1. 打开 [Amazon IVS 控制台](#)。

( 您还可通过[亚马逊云科技管理控制台](#)访问 Amazon IVS 控制台。 )

2. 在导航窗格中选择 Live channels ( 直播通道 )。( 如果导航窗格已折叠，请首先选择汉堡包图标以将其打开。 )
3. 选择 Filter by health ( 按运行状况筛选 ) 的搜索字段。
4. 在下拉列表中，选择筛选条件为 Health = STARVING ( 运行状况 = 匮乏 )。

筛选后，您可以转到通道的详细信息页面并选择该通道的实时流会话，以访问输入配置详细信息和流事件。

## CLI 说明

使用 Amazon CLI 是一个高级选项，需要先在计算机上下载并配置 CLI。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

按运行状况（例如 STARVING）筛选流：

```
aws ivs list-streams --filter-by health=STARVING
```

## ConcurrentStreams 的 CloudWatch 运行状况维度

您可以按特定 Health 筛选 ConcurrentStreams。请参阅 [CloudWatch 指标：IVS 低延迟直播功能](#)。

## 访问 CloudWatch 指标

Amazon CloudWatch 可从 Amazon IVS 收集和處理原始数据，并将数据处理为便于读取的近乎实时的指标。这些统计数据会保存 15 个月，从而使您能够从历史角度了解您的 Web 应用程序或服务的执行情况。您可以设置用于特定阈值的警报，并在达到相应阈值时发送通知或执行操作。有关详细信息，请参阅 [CloudWatch 用户指南](#)。

请注意，随着时间的推移，CloudWatch 指标会不断累积。随着指标的老化，分辨率实际有所降低。以下是计划：

- 1 秒的指标可用期为 3 小时。
- 60 秒的指标可用期为 15 天。
- 5 分钟的指标可用期为 63 天。
- 1 小时的指标可用 455 天（15 个月）。

当您调用 `getMetricData` 时，对于高分辨率指标，您可以将周期指定 1 秒、5 秒（推荐）、10 秒、30 秒或 60 秒的任意倍数。

## CloudWatch 控制台说明

1. 访问 <https://console.aws.amazon.com/cloudwatch/> 打开 CloudWatch 控制台。
2. 在侧导航栏中，展开 Metrics（指标）下拉菜单，然后选择 All metrics（所有指标）。

3. 在浏览选项卡上，使用左侧未标记的下拉菜单，选择您的“主”区域，即创建通道的区域。有关区域的详细信息，请参阅[全球解决方案](#)，[区域控制](#)。有关支持区域的列表，请参阅[亚马逊云科技一般参考](#)中的 Amazon IVS 页面。
4. 在浏览选项卡的底部，选择 IVS 命名空间。
5. 执行下列操作之一：
  - a. 在搜索栏中，输入资源 ID (是 ARN `arn:::ivs:channel/<resource id>` 的一部分)。

然后选择 IVS > 按通道。

- b. 如果 IVS 显示为 Amazon 命名空间下的一个可选服务，选择该服务。如果您使用 Amazon IVS 并将指标发送给 Amazon CloudWatch，则将会列出 IVS。(如果 IVS 未列出，则您没有任何 Amazon IVS 指标。)

然后根据需要选择维度分组；可用维度将在下面的 [CloudWatch 指标](#) 中列出。

6. 选择要添加到图表的指标。可用维度将在下面的 [CloudWatch 指标](#) 中列出。

您还可以从流会话的详细信息页面访问流会话的 CloudWatch 图表，方法是选择 View in CloudWatch (在 CloudWatch 中查看) 按钮。

## CLI 说明

您也可以使用 Amazon CLI 访问指标。这需要首先在计算机上下载并配置 CLI。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

然后，使用 Amazon CLI 访问 Amazon IVS 低延迟直播功能指标：

- 在命令提示符下，运行：

```
aws cloudwatch list-metrics --namespace AWS/IVS
```

有关更多信息，请参阅 Amazon CloudWatch 用户指南中的 [使用 Amazon CloudWatch 指标](#)。

## CloudWatch 指标：IVS 低延迟直播功能。

Amazon IVS 在 Amazon/IVS 命名空间中提供了以下指标。

指标	维度	描述
ConcurrentViews	无	<p>所有实时通道的并发视图计数。视图是一个独特的观看会话，它主动下载或播放视频。（有关更详细的定义，请参阅 <a href="#">术语表</a>。）如果通道处于活动状态，但总体上没有视图，则此指标的值为 0。如果没有实时通道，则此指标没有数据点。</p> <p>单位：计数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内并发视图的平均数、最大数或最小数（各自）。</p>
ConcurrentViews	Channel	<p>按通道 ARN 筛选 ConcurrentViews。如果有实时通道，但没有视图，则此指标的值为 0。如果通道未处于实时状态，则该指标没有数据点。</p> <p>此指标提供通道数据，而不是流数据。要查看给定通道上特定流传输会话的并发视图，请评估该通道在流传输会话开始和结束时间之间的 ConcurrentViews 指标。</p> <p>单位：计数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内并发视图的平均数、最大数或最小数（各自）。</p>
ConcurrentStreams	无	<p>正在进行流直播的通道计数。如果没有实时通道，则该指标没有数据点。</p> <p>单位：计数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内并发流的平均数、最大数或最小数（各自）。</p>

指标	维度	描述
ConcurrentStreams	Health	<p>按通道运行状况筛选 ConcurrentStreams 。如果没有实时通道，则该指标没有数据点。</p> <p>单位：计数</p> <p>有效统计数据：平均值、最大值、最小值，即在配置的时间间隔内特定 Health 的并发流的平均数、最大数或最小数（各自）。</p>
IngestAudioBitrate	Channel	<p>（高分辨率指标）Amazon IVS 在您流式传输时收到的音频数据量。更高的比特率会占用更多的可用互联网带宽。</p> <p>单位：每秒比特数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内接收音频比特率的平均数、最大数或最小数（各自）。</p>
IngestFrameRate	Channel	<p>（高分辨率指标）流式传输时，Amazon IVS 收到动画帧的频率。</p> <p>单位：每秒帧数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内接收帧率的平均数、最大数或最小数（各自）。</p>
IngestVideoBitrate	Channel	<p>（高分辨率指标）Amazon IVS 在您流式传输时收到的视频数据量。更高的比特率会占用更多的可用互联网带宽。更高的比特率可以提高视频质量，但只能达到一定程度。</p> <p>单位：每秒比特数</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内接收视频比特率的平均数、最大数或最小数（各自）。</p>

指标	维度	描述
KeyframeInterval	Channel	<p>( 高分辨率指标 ) 视频流中发送整个帧的点，而不仅仅是与上一帧的差异。</p> <p>单位：秒</p> <p>有效统计数据：平均值、最大值、最小值 – 在配置的时间间隔内关键帧的平均数、最大数或最小数 ( 各自 )。</p>
LiveDeliveredTime	无	<p>提供给所有查看者的视频的总实时持续时间。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>
LiveDeliveredTime	Channel	<p>按通道筛选 LiveDeliveredTime 。通道值是通道的 resource-id ，是 <a href="#">ARN</a> 的最后一部分。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>
LiveDeliveredTime	Channel, ViewerCountryCode	<p>按通道和查看者的国家代码筛选 LiveDeliveredTime 。通道值是通道的 resource-id ，即 <a href="#">ARN</a> 的最后一部分。国家/地区值是两个字符的 ISO 3166-1 国家/地区代码。这可以让您回答此问题：我的查看者从哪里观看？如果无法确定查看者的国家/地区，则显示为 UNKNOWN。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>

指标	维度	描述
LiveInputTime	无	<p>视频流的实时持续时间。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>
LiveInputTime	Channel	<p>按通道筛选 LiveInputTime 。通道值是通道的 resource-id ，是 <a href="#">ARN</a> 的最后一部分。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>
RecordedTime	无	<p>已录制视频的实时持续时间。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>
RecordedTime	Channel	<p>按通道筛选 RecordedTime 。通道值是通道的 resource-id ，是 <a href="#">ARN</a> 的最后一部分。</p> <p>单位：秒</p> <p>有效统计数据：总计</p>



## IVS 广播 SDK ( 低延迟流式传输 )

Amazon Interactive Video Services ( IVS ) 低延迟流式传输广播 SDK 适用于使用 Amazon IVS 构建应用程序的开发人员。此开发工具包旨在利用 Amazon IVS 架构，并将实现 Amazon IVS 的持续改进和新功能。作为本机广播开发工具包，它旨在最大限度地减少对应用程序以及用户有权访问应用程序所在设备的性能影响。

您的应用程序可以利用 Amazon IVS 广播开发工具包的主要功能：

- 高质量的流式传输 - 广播开发工具包支持高质量的流式传输。从您的相机捕获视频，并以高达 1080p 的质量对其进行编码，以获得高质量的观看体验。
- 自动比特率调整 - 智能手机用户是移动的，因此他们的网络条件会在整个广播过程中发生变化。Amazon IVS 广播开发工具包会自动调整视频比特率，以适应不断变化的网络条件。
- 支持纵向和横向 - 无论您的用户如何持有其设备，图像都会显示为顶部朝上并正确缩放。广播 SDK 支持纵向和横向画布大小。当用户从配置的方向旋转设备时，它会自动管理宽高比。
- 安全流式传输 - 使用 TLS 对用户的广播进行加密，因此他们可以保护其流的安全。
- 外部音频设备 - Amazon IVS 广播开发工具包支持音频插孔、USB 和蓝牙 SCO 外接麦克风。

## 平台要求

### 本机平台

平台	受支持的版本
Android	5.0 (Lollipop) 及更高版本
iOS	12 和更高版本  如果广播对于您的应用程序至关重要，请使用 <a href="#">UIRequiredDeviceCapabilities</a> ，指定 Metal 作为从 Apple 应用商店下载应用的要求。

IVS 至少支持 4 个主要 iOS 版本和 6 个主要 Android 版本。我们当前版本的支持可能会超出这些最低限度。如果主要版本不再受支持，将至少提前 3 个月通过 SDK 发布说明通知客户。

## 桌面浏览器

浏览器	支持的平台	受支持的版本
Chrome	Windows、macOS	两个主要版本（当前版本和最新版本）
Firefox	Windows、macOS	两个主要版本（当前版本和最新版本）
边缘	（Windows 8.1 和更高版本）	两个主要版本（当前版本和最新版本） 不包括 Edge Legacy
Safari	macOS	两个主要版本（当前版本和最新版本）

## 移动浏览器

浏览器	受支持的版本
iOS 版 Chrome、iOS 版 Safari	两个主要版本（当前版本和最新版本）
Chrome for iPadOS , Safari for iPadOS	两个主要版本（当前版本和最新版本）
Android 版 Chrome	两个主要版本（当前版本和最新版本）

## Webviews

Web 广播 SDK 不支持 Webviews 或 Weblike 环境（电视、控制台等）。有关移动实施，请参阅适用于 [Android](#) 和 [iOS](#) 的 Low-Latency Streaming Broadcast SDK Guide。

## 所需设备访问

广播开发工具包需要访问设备的摄像头和麦克风，包括设备内置的摄像头和麦克风以及通过蓝牙、USB 或音频插孔连接的摄像头和麦克风。

## 支持

如果您的视频流出现广播错误或其他问题，请通过广播 API 确定唯一的播放会话标识符。

对于此 Amazon IVS 广播开发工具包：	使用：
Android	BroadcastSession 上的 getSessionId 函数
iOS	IVSBroadcastSession 的属性 sessionId
Web	getSessionId 函数

与 Amazon Support 共享此广播会话标识符。利用它可获得有助于解决问题的信息。

注意：广播开发工具包在不断改进。请参阅 [Amazon IVS 发布说明](#) 了解可用版本和已修复问题。如果合适，请在联系支持部门之前更新您的广播开发工具包版本，看看这是否解决了您的问题。

## 版本控制

Amazon IVS 广播开发工具包使用 [语义化版本](#)。

在此讨论中，假设：

- 最新版本是 4.1.3。
- 先前主要版本的最新版本为 3.2.4。
- 版本 1.x 最新版本是 1.5.6。

最新版本的次要版本已添加向后兼容的新功能。在本例中，版本 4.2.0 已添加新功能。

最新版本的补丁版本已添加向后兼容、次要错误修复。在这里，版本 4.1.4 已添加次要错误修复。

向后兼容、主要错误修复处理方式不同；将在以下几个版本中添加：

- 最新版本补丁版本。在本例中是版本 4.1.4。
- 先前次要版本的补丁版本。在本例中是版本 3.2.5。
- 最新版本 1.x 版本的补丁版本。在本例中是版本 1.5.7。

主要错误修复由 Amazon IVS 产品团队定义。典型示例包括关键安全更新和客户所需的其他选定修复。

注意：在上面的例子中，发布的版本递增但不会跳过任何数字（例如，从 4.1.3 到 4.1.4）。实际上，一个或多个补丁编号可能保留在内部而不发布，因此发布版本可以从 4.1.3 增加到 4.1.6。

## IVS Broadcast SDK: Web Guide (低延迟流式传输)

IVS 低延迟流式传输 Web 广播 SDK 为开发人员提供了在 Web 上构建交互式实时体验的工具。

最新版本的网络广播 SDK：1.8.0 ([版本说明](#))

参考文档：有关亚马逊 IVS 网络广播软件开发工具包中可用的最重要方法的信息，请参阅 <https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference>。请确保选择最新版本的 SDK。

示例代码：以下示例可以帮助您快速开始使用 SDK：

- [向 IVS 频道进行单次广播 \(HTML 和 JavaScript\)](#)
- [将屏幕共享到 IVS 通道的单个广播 \(React 源代码\)](#)

平台要求：有关支持的平台列表，请参阅 [Amazon IVS 广播 SDK](#)。

## 开始使用

### 安装库

请注意，IVS BroadcastClient 利用[反射元数据](#)，它扩展了全局 `Reflect` 对象。尽管这不会造成任何冲突，但在极少数情况下，这可能会导致不必要的行为。

### 使用脚本标签

网络广播 SDK 作为 JavaScript 库分发，可通过 <https://web-broadcast.live-video.net/1.8.0/amazon-ivs-web-broadcast.js> 进行检索。

通过 `<script>` 标签加载时，该库会在窗口作用域中公开一个名为 `IVSBroadcastClient` 的全局变量。

### 使用 npm

安装 npm 程序包：

```
npm install amazon-ivs-web-broadcast
```

现在，您可以访问 `IVSBroadcastClient` 对象并拉入其他模块和常量，例如 `Errors`、`BASIC_LANDSCAPE`：

```
import IVSBroadcastClient, {
  Errors,
  BASIC_LANDSCAPE
} from 'amazon-ivs-web-broadcast';
```

## 样本

要快速入门，请参阅以下示例：

- [向 IVS 频道进行单次广播 \(HTML 和 JavaScript\)](#)
- [将屏幕共享到 IVS 通道的单个广播 \(React 源代码\)](#)

## 创建 AmazonIVS 的实例 BroadcastClient

要使用该库，您必须创建一个客户端实例。为此，您可以使用 `streamConfig` 参数（指定广播的限制，如分辨率和帧率）调用 `IVSBroadcastClient` 的 `create` 方法。您可以在创建客户端时指定摄取端点，也可以在启动视频流时设置此端点。

采集终端节点可以在 AWS 控制台中找到，也可以由终端节点返回（例如 `unique_id.global-Contribute.Live-video.net CreateChannel t`）。

```
const client = IVSBroadcastClient.create({
  // Enter the desired stream configuration
  streamConfig: IVSBroadcastClient.BASIC_LANDSCAPE,
  // Enter the ingest endpoint from the AWS console or CreateChannel API
  ingestEndpoint: 'UNIQUE_ID.global-contribute.live-video.net',
});
```

以下是常见的受支持的流配置。预设为高达 480p 和 1.5 Mbps 比特率的 BASIC 配置、高达 1080p 和 3.5 Mbps 比特率的 BASIC 全高清配置，以及高达 1080p 和 8.5 Mbps 比特率的 STANDARD（或 ADVANCED）配置。您可以根据需要自定义比特率、帧率和分辨率。有关更多信息，请参阅[BroadcastClientConfig](#)。

```
IVSBroadcastClient.BASIC_LANDSCAPE;
```

```
IVSBroadcastClient.BASIC_FULL_HD_LANDSCAPE;  
IVSBroadcastClient.STANDARD_LANDSCAPE;  
IVSBroadcastClient.BASIC_PORTRAIT;  
IVSBroadcastClient.BASIC_FULL_HD_PORTRAIT;  
IVSBroadcastClient.STANDARD_PORTRAIT;
```

如果使用 npm 程序包，可以单独导入这些配置。

注意：确保您的客户端配置与后端通道类型一致。例如，如果通道类型为 STANDARD，则 streamConfig 应设置为 IVSBroadcastClient.STANDARD\_\* 值中的一个。如果通道类型为 ADVANCED，则需要手动设置配置，如下所示（以 ADVANCED\_HD 为例）：

```
const client = IVSBroadcastClient.create({  
  // Enter the custom stream configuration  
  streamConfig: {  
    maxResolution: {  
      width: 1080,  
      height: 1920,  
    },  
    maxFramerate: 30,  
    /**  
     * maxBitrate is measured in kbps  
     */  
    maxBitrate: 3500,  
  },  
  // Other configuration . . .  
});
```

## 请求权限

您的应用程序必须请求权限才能访问用户的摄像头和麦克风，并且必须使用 HTTPS 发送请求。（这不是 Amazon IVS 特有的；需要访问摄像头和麦克风的任何网站都需要请求权限。）

以下示例函数显示了如何请求和获取音频和视频设备的权限：

```
async function handlePermissions() {  
  let permissions = {  
    audio: false,  
    video: false,  
  };  
  try {  
    const stream = await navigator.mediaDevices.getUserMedia({ video: true, audio:  
true });
```

```
    for (const track of stream.getTracks()) {
      track.stop();
    }
    permissions = { video: true, audio: true };
  } catch (err) {
    permissions = { video: false, audio: false };
    console.error(err.message);
  }
  // If we still don't have permissions after requesting them display the error
  message
  if (!permissions.video) {
    console.error('Failed to get video permissions.');
```

有关更多信息，请参阅[权限 API](#) 和[MediaDevices.getUserMedia\(\)](#)。

## 设置流式预览

要预览将要播放的内容，请为 SDK 提供一个 `<canvas>` 元素。

```
// where #preview is an existing <canvas> DOM element on your page
const previewEl = document.getElementById('preview');
client.attachPreview(previewEl);
```

## 列出可用的设备

要查看哪些设备可供捕获，请查询浏览器的 [MediaDevices.enumerateDevices\(\)](#) 方法：

```
const devices = await navigator.mediaDevices.enumerateDevices();
window.videoDevices = devices.filter((d) => d.kind === 'videoinput');
window.audioDevices = devices.filter((d) => d.kind === 'audioinput');
```

## MediaStream 从设备中检索

获取可用设备列表后，您可以从任意数量的设备中检索媒体流。例如，您可以使用 `getUserMedia()` 方法从摄像头中检索媒体流。

如果您想指定从哪个设备捕获媒体流，可以在媒体限制的 `audio` 或 `video` 部分明确设置 `deviceId`。或者，您可以省略 `deviceId`，让用户从浏览器提示中选择他们的设备。

您还可以使用 `width` 和 `height` 限制来指定理想的摄像头分辨率。（请在[此处](#)阅读有关这些限制的更多信息。） SDK 会自动应用与您的最大广播分辨率相对应的宽度和高度限制；但是，最好自己也应用这些限制，以确保将源添加到 SDK 后源宽高比不会改变。

```
const streamConfig = IVSBroadcastClient.BASIC_LANDSCAPE;
...
window.cameraStream = await navigator.mediaDevices.getUserMedia({
  video: {
    deviceId: window.videoDevices[0].deviceId,
    width: {
      ideal: streamConfig.maxResolution.width,
    },
    height: {
      ideal: streamConfig.maxResolution.height,
    },
  },
});
window.microphoneStream = await navigator.mediaDevices.getUserMedia({
  audio: { deviceId: window.audioDevices[0].deviceId },
});
```

## 将设备添加到流中

获取流后，您可以通过指定唯一的名称（下面的 `camera1`）和合成位置（适用于视频）将设备添加到布局中。例如，通过指定网络摄像头设备，您可以将网络摄像头视频源添加到广播流中。

指定视频输入设备时，必须指定索引，索引表示要进行广播的“层”。这是图像编辑或 CSS 的同义词，其中 `z` 索引表示要渲染的层的顺序。或者，您可以提供一个位置，该位置定义了流源的 `x/y` 坐标（以及大小）。

有关参数的详细信息，请参阅[VideoComposition](#)。

```
client.addVideoInputDevice(window.cameraStream, 'camera1', { index: 0 }); // only
  'index' is required for the position parameter
client.addAudioInputDevice(window.microphoneStream, 'mic1');
```

## 开启广播

要开始广播，请提供您的 Amazon IVS 通道的流密钥：

```
client
```



```
.startBroadcast(streamKey)
.then((result) => {
    console.log('I am successfully broadcasting!');
})
.catch((error) => {
    console.error('Something drastically failed while broadcasting!', error);
});
```

## 停止广播

```
client.stopBroadcast();
```

## 交换视频位置

客户端支持交换视频设备的合成位置：

```
client.exchangeVideoDevicePositions('camera1', 'camera2');
```

## Mute Audio

要将音频静音，请使用 `removeAudioInputDevice` 删除音频设备，或者在音轨上设置 `enabled` 属性：

```
let audioStream = client.getAudioInputDevice(AUDIO_DEVICE_NAME);
audioStream.getAudioTracks()[0].enabled = false;
```

其中 `AUDIO_DEVICE_NAME` 是在 `addAudioInputDevice()` 调用过程中为原始音频设备指定的名称。

要取消静音：

```
let audioStream = client.getAudioInputDevice(AUDIO_DEVICE_NAME);
audioStream.getAudioTracks()[0].enabled = true;
```

## 隐藏视频

要隐藏视频，请使用 `removeVideoInputDevice` 删除视频设备，或者在视频轨道上设置 `enabled` 属性：

```
let videoStream = client.getVideoInputDevice(VIDEO_DEVICE_NAME).source;
```

```
videoStream.getVideoTracks()[0].enabled = false;
```

其中 VIDEO\_DEVICE\_NAME 是在原始 addVideoInputDevice() 调用过程中为视频设备指定的名称。

要取消隐藏：

```
let videoStream = client.getVideoInputDevice(VIDEO_DEVICE_NAME).source;  
videoStream.getVideoTracks()[0].enabled = true;
```

## 已知问题和解决方法

- 网络广播 SDK 需要打开端口 4443。VPN 和防火墙可能会屏蔽端口 4443 并阻止您使用直播功能。

解决办法：禁用 VPN 和/或配置防火墙以确保端口 4443 不会被屏蔽。

- 从横向模式切换到纵向模式有问题。

解决办法：尚无。

- HLS 清单中报告的分辨率不正确。它被设置为最初收到的分辨率，该分辨率通常远低于可能的分辨率，并且不反映在 webRTC 连接期间进行的任何规模升级。

解决办法：尚无。

- 加载初始页面后创建的后续客户端实例可能不会响应与第一个客户端实例不同的 maxFramerate 设置。

解决办法：创建第一个客户端实例时，通过 IVSBroadcastClient.create 函数仅设置一次 StreamConfig。

- 在 iOS 上，不支持捕获多个视频设备源 WebKit。

解决办法：关注[此问题](#)以跟踪开发进度。

- 在 iOS 上，一旦您已经有了视频源，调用 getUserMedia() 将停止通过 getUserMedia() 检索的任何其他视频源。

解决办法：尚无。

- WebRTC 为可用资源动态选择最佳比特率和分辨率。如果您的硬件或网络不支持 WebRTC，视频流的质量就不会很高。在广播过程中，视频流的质量可能会随着可用资源的增加或减少而变化。

解决办法：提供至少 200 kbps 的上传速度。

- 如果某个通道启用了 Auto-Record to Amazon S3 ( 自动录制到 Amazon S3 )，并且使用了 Web 广播 SDK，则录制到相同的 S3 前缀可能不起作用，因为 Web 广播 SDK 会动态更改比特率和质量。

解决办法：尚无。

- 使用 Next.js 时，可能会遇到 Uncaught ReferenceError: self is not defined 错误，具体取决于 SDK 的导入方式。

解决方法：使用 Next.js 时[动态导入库](#)。

- 您可能无法使用 module 类型的脚本标签 ( 即 `<script type="module" src="...">` ) 导入模块。

解决方法：该库没有 ES6 构建版本。从脚本标签中删除 `type="module"`。

- Safari 广播的观众有时会在视频源中看到绿色构件。

解决方法：重新初始化用户媒体或重新启动广播。

## Safari 限制

- 拒绝权限提示需要在操作系统级别重置 Safari 网站设置中的权限。
- Safari 本身无法像 Firefox 或 Chrome 那样有效地检测所有设备。例如，未检测到 OBS 虚拟摄像头。

## Firefox 限制

- Firefox 屏幕共享需要启用系统权限。启用它们后，用户必须重新启动 Firefox 才能正常运行；否则，如果权限被视为被阻止，浏览器将抛出 `NotFoundError` 异常。
- 缺少 `getCapabilities` 方法。这意味着用户无法获得媒体轨道的分辨率或宽高比。请参阅此 [bugzilla 主题帖](#)。
- 缺少几个 `AudioContext` 属性；例如，延迟和通道数。这可能会给想要操作音轨的高级用户带来问题。
- 在 MacOS 上，来自 `getUserMedia` 的摄像头画面被限制为 4:3 的宽高比。请参阅 [bugzilla 主题帖 1](#) 和 [bugzilla 主题帖 2](#)。
- `getDisplayMedia` 不支持音频捕获。请参阅此 [bugzilla 主题帖](#)。
- 屏幕捕获的帧率不理想（大约 15fps？）。请参阅此 [bugzilla 主题帖](#)。

## IVS Broadcast SDK: Android Guide ( 低延迟流式传输 )

IVS 低延迟流式传输 Android 广播 SDK 提供了在 Android 上广播到 IVS 所需的接口。

`com.amazonaws.ivs.broadcast` 软件包实现了本文档中所描述的接口。支持以下操作：

- 设置（初始化）广播会话。
- 管理广播。
- 连接和分离输入设备。
- 管理合成会话。
- 接收事件。
- 接收错误。

最新版本的安卓广播 SDK：1.14.1 ( [发布说明](#) )

参考文档：有关亚马逊 IVS 安卓广播 SDK 中可用的最重要方法的信息，请参阅参考文档 <https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/android/>。

示例代码：参见 Android 示例存储库，[网址为 GitHub：https://github.com/aws-samples/amazon-ivs-broadcast-android-sample](https://github.com/aws-samples/amazon-ivs-broadcast-android-sample)。

平台要求：Android 5.0 ( Lollipop ) 或更高版本

## 开始使用

### 安装库

要将 Amazon IVS Android 广播库添加到您的 Android 开发环境中，请将该库添加到您模块的 `build.gradle` 文件，如此处所示（适用于最新版本的 Amazon IVS 广播开发工具包）。

```
repositories {
    mavenCentral()
}

dependencies {
    implementation 'com.amazonaws:ivs-broadcast:1.14.1'
}
```

如要手动安装 SDK，也可从以下位置下载最新版本：

<https://search.maven.org/artifact/com.amazonaws/ivs-broadcast>

### 创建事件侦听器

设置事件侦听器可接收状态更新、设备更改通知、错误和会话音频信息。

```
BroadcastSession.Listener broadcastListener =
    new BroadcastSession.Listener() {
        @Override
        public void onStateChanged(@NonNull BroadcastSession.State state) {
            Log.d(TAG, "State=" + state);
        }

        @Override
        public void onError(@NonNull BroadcastException exception) {
            Log.e(TAG, "Exception: " + exception);
        }
    };
```

## 请求权限

您的应用必须请求权限才能访问用户摄像头和麦克风。（这并非特定于 Amazon IVS；需要访问摄像头和麦克风的任何应用程序都需要这样做。）

我们在此处检查用户是否已授予权限，如果没有，对他们提出要求：

```
final String[] requiredPermissions =
    { Manifest.permission.CAMERA, Manifest.permission.RECORD_AUDIO };

for (String permission : requiredPermissions) {
    if (ContextCompat.checkSelfPermission(this, permission)
        != PackageManager.PERMISSION_GRANTED) {
        // If any permissions are missing we want to just request them all.
        ActivityCompat.requestPermissions(this, requiredPermissions, 0x100);
        break;
    }
}
```

在这里，我们得到用户的响应：

```
@Override
public void onRequestPermissionsResult(int requestCode,
                                       @NonNull String[] permissions,
                                       @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode,
                                     permissions, grantResults);
    if (requestCode == 0x100) {
        for (int result : grantResults) {
            if (result == PackageManager.PERMISSION_DENIED) {
                return;
            }
        }
        setupBroadcastSession();
    }
}
```

## 创建广播会话

广播接口为 `com.amazonaws.ivs.broadcast.BroadcastSession`。使用预设值对它进行初始化，如下所示。如果在初始化过程中出现任何错误（例如配置编解码器失败），则 `BroadcastListener` 将得到错误消息，并且 `broadcastSession.isReady` 将变为 `false`。

**重要提示：**对 Amazon IVS Broadcast SDK for Android 的所有调用必须在实例化该开发工具包的线程上进行。来自不同线程的调用将导致开发工具包抛出致命错误并停止广播。

```
// Create a broadcast-session instance and sign up to receive broadcast
// events and errors.
Context ctx = getApplicationContext();
broadcastSession = new BroadcastSession(ctx,
    broadcastListener,
    Presets.Configuration.STANDARD_PORTRAIT,
    Presets.Devices.FRONT_CAMERA(ctx));
```

另请参阅[创建广播会话 \(高级版\)](#)。

## 设置 ImagePreviewView 为预览

如果要显示活动摄像机设备的预览，请将设备的预览 ImagePreviewView 添加到您的视图层次结构。

```
// awaitDeviceChanges will fire on the main thread after all pending devices
// attachments have been completed
broadcastSession.awaitDeviceChanges(() -> {
    for(Device device: session.listAttachedDevices()) {
        // Find the camera we attached earlier
        if(device.getDescriptor().type == Device.Descriptor.DeviceType.CAMERA) {
            LinearLayout previewHolder = findViewById(R.id.previewHolder);
            ImagePreviewView preview = ((ImageDevice)device).getPreviewView();
            preview.setLayoutParams(new LinearLayout.LayoutParams(
                LinearLayout.LayoutParams.MATCH_PARENT,
                LinearLayout.LayoutParams.MATCH_PARENT));
            previewHolder.addView(preview);
        }
    }
});
```

## 开启广播

您在 GetChannel 端点的 ingestEndpoint 响应字段中接收的主机名的前面应为 `rtmps://`，后面为 `/app`。完整 URL 应采用以下格式：`rtmps://{ ingestEndpoint }/app`

```
broadcastSession.start(IVS_RTMP_URL, IVS_STREAMKEY);
```

Android 广播 SDK 仅支持 RTMPS 采集 ( 不支持不安全的 RTMP 采集 ) 。

## 停止广播

```
broadcastSession.stop();
```

## 发布广播会话

当不再使用广播会话时，必须调用 `broadcastSession.release()` 方法，以释放库使用的资源。

```
@Override
protected void onDestroy() {
    super.onDestroy();
    previewHolder.removeAllViews();
    broadcastSession.release();
}
```

## 高级使用案例

我们将在此处介绍一些高级使用案例。从上面的基本设置开始，然后在此处继续。

### 创建广播配置

我们将在此处创建一个带有两个混合器插槽的自定义配置，允许我们将两个视频源绑定到混合器。一个 (custom) 是全屏幕并布置在另一个 (camera) 后面，它更小且位于右下角。请注意，对于 custom 插槽，我们不设置位置、大小或宽高比模式。因为我们不设置这些参数，所以插槽将使用视频设置的大小和位置。

```
BroadcastConfiguration config = BroadcastConfiguration.with($ -> {
    $.audio.setBitrate(128_000);
    $.video.setMaxBitrate(3_500_000);
    $.video.setMinBitrate(500_000);
    $.video.setInitialBitrate(1_500_000);
    $.video.setSize(1280, 720);
    $.mixer.slots = new BroadcastConfiguration.Mixer.Slot[] {
        BroadcastConfiguration.Mixer.Slot.with(slot -> {
            // Do not automatically bind to a source
            slot.setPreferredAudioInput(
                Device.Descriptor.DeviceType.UNKNOWN);
            // Bind to user image if unbound
            slot.setPreferredVideoInput(
                Device.Descriptor.DeviceType.USER_IMAGE);
        });
    };
});
```



```

        slot.setName("custom");
        return slot;
    }},
    BroadcastConfiguration.Mixer.Slot.with(slot -> {
        slot.setzIndex(1);
        slot.setAspect(BroadcastConfiguration.AspectMode.FILL);
        slot.setSize(300, 300);
        slot.setPosition($.video.getSize().x - 350,
            $.video.getSize().y - 350);
        slot.setName("camera");
        return slot;
    })
};
return $;
});

```

## 创建广播会话 (高级版)

就像在[基本示例](#)中一样创建 `BroadcastSession`，但在此处会提供您的自定义配置。还对设备阵列提供 `null`，因为我们将手动添加它们。

```

// Create a broadcast-session instance and sign up to receive broadcast
// events and errors.
Context ctx = getApplicationContext();
broadcastSession = new BroadcastSession(ctx,
    broadcastListener,
    config, // The configuration we created above
    null); // We'll manually attach devices after

```

## 迭代和连接摄像机设备

我们将通过开发工具包检测到的输入设备进行迭代。在 Android 7 (Nougat) 上，这将仅返回默认麦克风设备，因为 Amazon IVS 广播开发工具包不支持在此版本的 Android 上选择非默认设备。

找到我们想要使用的设备之后，我们会调用 `attachDevice` 来连接它。连接输入设备完成后，在主线程上调用 `lambda` 函数。如果出现故障，您将在侦听器中收到错误。

```

for(Device.Descriptor desc:
    BroadcastSession.listAvailableDevices(getApplicationContext())) {
    if(desc.type == Device.Descriptor.DeviceType.CAMERA &&
        desc.position == Device.Descriptor.Position.FRONT) {
        session.attachDevice(desc, device -> {

```

```
        LinearLayout previewHolder = findViewById(R.id.previewHolder);
        ImagePreviewView preview = ((ImageDevice)device).getPreviewView();
        preview.setLayoutParams(new LinearLayout.LayoutParams(
            LinearLayout.LayoutParams.MATCH_PARENT,
            LinearLayout.LayoutParams.MATCH_PARENT));
        previewHolder.addView(preview);
        // Bind the camera to the mixer slot we created above.
        session.getMixer().bind(device, "camera");
    });
    break;
}
}
```

## 交换摄像机

```
// This assumes you've kept a reference called "currentCamera" that points to
// a front facing camera
for(Device device: BroadcastSession.listAvailableDevices()) {
    if(device.type == Device.Descriptor.DeviceType.CAMERA &&
        Device.position != currentCamera.position) {
        // Remove the preview view for the old device.
        // setImagePreviewTextureView is an example function
        // that handles your view hierarchy.
        setImagePreviewView(null);
        session.exchangeDevices(currentCamera, device, camera -> {
            // Set the preview view for the new device.
            setImagePreviewView(camera.getPreviewView());
            currentCamera = camera;
        });
        break;
    }
}
```

## 创建输入表面

若要输入应用程序生成的声音或图像数据，请使用 `createImageInputSource` 或者 `createAudioInputSource`。这两种方法都会创建和连接虚拟设备，这些设备可以像任何其他设备一样绑定到混合器。

`createImageInputSource` 返回的 `SurfaceSource` 具有 `getInputSurface` 方法，它会为您提供一个 `Surface`，可以将其与 `Camera2 API`、`OpenGL` 或 `Vulkan` 或可以写入到 `Surface` 的任何其他接口一起使用。

AudioDevice返回的createAudioInputSource可以接收通过 AudioRecorder 或其他方式生成的线性 PCM 数据。

```
SurfaceSource source = session.createImageInputSource();
Surface surface = source.getInputSurface();
session.getMixer().bind(source, "custom");
```

## 分离设备

如果要分离而不是替换设备，请使用 Device 或 Device.Descriptor 来分离它。

```
session.detachDevice(currentCamera);
```

## 屏幕和系统音频捕获

Amazon IVS Broadcast SDK for Android 含有一些帮助程序，可以简化捕获设备屏幕（Android 5 及更高版本）和系统音频（Android 10 及更高版本）。如果要手动管理这些数据，可以创建自定义图像输入源和自定义音频输入源。

若要创建屏幕和系统音频捕获会话，您必须首先创建权限请求目的：

```
public void startScreenCapture() {
    MediaProjectionManager manager =
        (MediaProjectionManager) getApplicationContext()
            .getSystemService(Context.MEDIA_PROJECTION_SERVICE);
    if(manager != null) {
        Intent intent = manager.createScreenCaptureIntent();
        startActivityIfNeeded(intent, SCREEN_CAPTURE_REQUEST_ID);
    }
}
```

若要使用此功能，必须提供扩展 com.amazonaws.ivs.broadcast.SystemCaptureService 的类。您不必覆盖其任何方法，但需要使用类以避免服务之间的任何潜在冲突。

您还必须向 Android 清单添加几个元素：

```
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
<application ...>
    <service android:name=".ExampleSystemCaptureService"
        android:foregroundServiceType="mediaProjection"
        android:isolatedProcess="false" />
```

```
</application>
...
```

必须在 `<service>` 元素内命名扩展 `SystemCaptureService` 的类。在 Android 9 及更高版本上，`foregroundServiceType` 必须为 `mediaProjection`。

返回权限目的之后，您可以继续创建屏幕和系统音频捕获会话。在 Android 8 及更高版本中，您必须提供要显示在用户的通知面板中的通知。Amazon IVS Broadcast SDK for Android 提供了便利的方法 `createServiceNotificationBuilder`。或者，您也可以提供自己的通知。

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode != SCREEN_CAPTURE_REQUEST_ID
        || Activity.RESULT_OK != resultCode) {
        return;
    }
    Notification notification = null;
    if(Build.VERSION.SDK_INT >= 26) {
        Intent intent = new Intent(getApplicationContext(),
                                    NotificationActivity.class);

        notification = session
            .createServiceNotificationBuilder("example",
                                             "example channel", intent)
            .build();
    }
    session.createSystemCaptureSources(data,
                                       ExampleSystemCaptureService.class,
                                       notification,
                                       devices -> {
                                           // This step is optional if the mixer slots have been given preferred
                                           // input device types SCREEN and SYSTEM_AUDIO
                                           for (Device device : devices) {
                                               session.getMixer().bind(device, "game");
                                           }
                                       });
}
```

## 获取推荐的广播设置

若要在开始广播之前评估用户的连接，请使用 `recommendedVideoSettings` 方法来运行一个简短的测试。在测试运行时，您将收到多个建议，从最推荐到最不推荐的顺序排列。在此版本的开发工具

包中，无法重新配置当前 `BroadcastSession`，因此您需要 `release()`，然后使用推荐的设置创建一个新的。您将继续收到 `BroadcastSessionTest.Results`，直到 `Result.status` 为 `SUCCESS` 或者 `ERROR`。您可以使用 `Result.progress` 检查进度。

Amazon IVS 支持 8.5 Mbps 的最大比特率（对于其 `type` 为 `STANDARD` 或 `ADVANCED` 的通道），所以此方法返回的 `maximumBitrate` 永远不会超过 8.5 Mbps。考虑到网络性能的小波动，建议此方法返回的 `initialBitrate` 略低于测试中测量的真实比特率。（通常不建议使用 100% 的可用带宽。）

```
void runBroadcastTest() {
    this.test = session.recommendedVideoSettings(RTMP_ENDPOINT, RTMP_STREAMKEY,
        result -> {
            if (result.status == BroadcastSessionTest.Status.SUCCESS) {
                this.recommendation = result.recommendations[0];
            }
        });
}
```

## 使用蓝牙麦克风

要使用蓝牙麦克风设备进行广播，必须启动蓝牙 SCO 连接：

```
Bluetooth.startBluetoothSco(context);
// Now bluetooth microphones can be used
...
// Must also stop bluetooth SCO
Bluetooth.stopBluetoothSco(context);
```

## 已知问题和解决方法

- 使用通过蓝牙而连接的外部麦克风可能不稳定。当蓝牙设备在广播会话期间连接或断开连接时，麦克风输入可能会停止工作，直到设备明确分离并重新连接。

解决办法：如果您计划使用蓝牙耳机，请在开始广播之前连接它，并在整个广播期间保持连接状态。

- 广播开发工具包不支持通过 USB 连接的外部摄像机访问。

解决办法：请勿使用通过 USB 连接的外部摄像机。

- 提交音频数据的速度比实时更快（使用自定义音频源）会导致音频偏差。

解决办法：不要以快于实时的速度提交音频数据。

- 如果相同的 BroadcastSession 用于多个广播，一些 Android 5 设备可能会流式传输黑色图像。  
解决办法：停止 BroadcastSession 时，释放它并实例化一个新的。
- Android 5、6 和 7 设备无法接收广播开发工具包麦克风的 onDeviceAdded 和 onDeviceRemoved 回调，因为这些 Android 版本只允许系统的默认麦克风。  
解决办法：对于这些设备，广播开发工具包使用系统的默认麦克风。
- 当从父级中删除 ImagePreviewView 时（例如，在父级调用 removeView()），会立即释放 ImagePreviewView。将其添加到另一个父视图时，ImagePreviewView 不显示任何帧。  
解决方法：使用 getPreview 请求再次预览。
- 某些 Android 视频编码器不能配置小于 176x176 的视频大小。配置较小的尺寸会导致错误并阻止流式传输。  
解决办法：请勿将视频大小配置为小于 176x176。

## IVS Broadcast SDK: iOS Guide ( 低延迟流式传输 )

IVS 低延迟流式传输 iOS 广播 SDK 提供了在 iOS 上广播到 Amazon IVS 所需的接口。

AmazonIVSBroadcast 模块实施了本文档中所描述的接口。支持以下操作：

- 设置（初始化）广播会话。
- 管理广播。
- 连接和分离输入设备。
- 管理合成会话。
- 接收事件。
- 接收错误。

最新版本的 iOS 广播 SDK：1.14.1 ( [版本说明](#) )

参考文档：有关亚马逊 IVS iOS 广播 SDK 中可用的最重要方法的信息，请参阅参考文档 <https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/ios/>。

示例代码：参见 iOS 示例存储库，[网址为 GitHub：https://github.com/aws-samples/amazon-ivs-broadcast-ios-sample](https://github.com/aws-samples/amazon-ivs-broadcast-ios-sample)。

平台要求：iOS 12 或更高版本

## 开始使用

### 安装库

我们建议您通过集成广播 SDK CocoaPods。（或者，您可以手动将框架添加至项目。）

推荐：集成广播 SDK (CocoaPods)

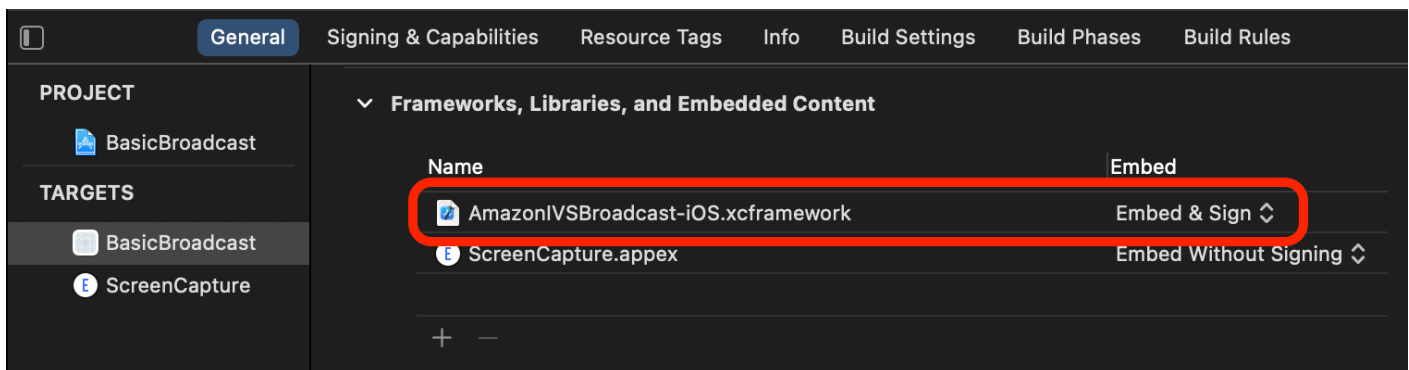
发行版以该 CocoaPods 名称发布 AmazonIVSBroadcast。将此依赖项添加至您的 Podfile 中：

```
pod 'AmazonIVSBroadcast'
```

运行 `pod install`，开发工具包将在 `.xcworkspace` 中可用。

替代方法：手动安装框架

1. 从 <https://broadcast.live-video.net/1.14.1/AmazonIVSBroadcast.xcframework.zip> 下载最新版本。
2. 提取归档的内容。AmazonIVSBroadcast.xcframework 包含适用于设备和模拟器的开发工具包。
3. 通过以下方法嵌入 AmazonIVSBroadcast.xcframework：将其拖动到应用程序目标的 General（常规）选项卡中的 Frameworks, Libraries, and Embedded Content（框架、库和嵌入式内容）部分中。



### 实施 IVS BroadcastSession .Delegate

实施 `IVSBroadcastSession.Delegate`，它允许您接收状态更新和设备更改通知：

```
extension ViewController : IVSBroadcastSession.Delegate {
    func broadcastSession(_ session: IVSBroadcastSession,
                        didChange state: IVSBroadcastSession.State) {
        print("IVSBroadcastSession did change state \(state)")
    }
}
```

```
}  
  
func broadcastSession(_ session: IVSBroadcastSession,  
                    didEmitError error: Error) {  
    print("IVSBroadcastSession did emit error \(error)")  
}  
}
```

## 请求权限

您的应用必须请求权限才能访问用户摄像头和麦克风。（这并非特定于 Amazon IVS；需要访问摄像头和麦克风的任何应用程序都需要这样做。）

我们在此处检查用户是否已授予权限，如果没有，对他们提出要求：

```
switch AVCaptureDevice.authorizationStatus(for: .video) {  
case .authorized: // permission already granted.  
case .notDetermined:  
    AVCaptureDevice.requestAccess(for: .video) { granted in  
        // permission granted based on granted bool.  
    }  
case .denied, .restricted: // permission denied.  
@unknown default: // permissions unknown.  
}
```

如果您希望分别访问摄像头和麦克风，则需要对 `.video` 和 `.audio` 媒体类型进行此操作。

您还需要将 `NSCameraUsageDescription` 和 `NSMicrophoneUsageDescription` 的条目添加到 `Info.plist`。否则，尝试请求权限时，您的应用程序将崩溃。

## 禁用应用程序空闲计时器

您可以自由选择，但我们建议您这样做。它可以防止您的设备在使用广播开发工具包时进入睡眠状态，这会中断广播。

```
override func viewDidAppear(_ animated: Bool) {  
    super.viewDidAppear(animated)  
    UIApplication.shared.isIdleTimerDisabled = true  
}  
  
override func viewDidDisappear(_ animated: Bool) {  
    super.viewDidDisappear(animated)  
    UIApplication.shared.isIdleTimerDisabled = false  
}
```



```
}
```

## ( 可选 ) 设置 AV AudioSession

默认情况下，广播开发工具包将设置您的应用程序的 AVAudioSession。如果您想自己管理此工具包，请将 `IVSBroadcastSession.applicationAudioSessionStrategy` 设置为 `noAction`。如果不控制 AVAudioSession，广播开发工具包无法在内部管理麦克风。要将麦克风与 `noAction` 选项结合使用，您可以创建 `IVSCustomAudioSource` 并通过 `AVCaptureSession`、`AVAudioEngine` 或提供 PCM 音频样本的其他工具提供您自己的样本。

如果您手动设置 AVAudioSession，您至少需要将类别设置为 `.record` 或者 `.playbackAndRecord`，并将其设置为 `active`。如果要从蓝牙设备录制音频，还需要指定 `.allowBluetooth` 选项：

```
do {
    try AVAudioSession.sharedInstance().setCategory(.record, options: .allowBluetooth)
    try AVAudioSession.sharedInstance().setActive(true)
} catch {
    print("Error configuring AVAudioSession")
}
```

我们建议让开发工具包为您处理此问题。否则，如果要在不同的音频设备之间进行选择，则需要手动管理这些端口。

## 创建广播会话

广播接口为 `IVSBroadcastSession`。将其初始化，如下所示：

```
let broadcastSession = try IVSBroadcastSession(
    configuration: IVSPresets.configurations().standardLandscape(),
    descriptors: IVSPresets.devices().frontCamera(),
    delegate: self)
```

另请参阅 [创建广播会话 \(高级版\)](#)

## 将 IVS 设置 ImagePreviewView 为预览

如果要显示活动摄像机设备的预览，请将设备的预览 `IVSImagePreviewView` 添加到您的视图层次结构：

```
// If the session was just created, execute the following
```

```
// code in the callback of IVSBroadcastSession.awaitDeviceChanges
// to ensure all devices have been attached.
if let devicePreview = try broadcastSession.listAttachedDevices()
    .compactMap({ $0 as? IVSImageDevice })
    .first?
    .previewView()
{
    previewView.addSubview(devicePreview)
}
```

## 开启广播

您在 GetChannel 端点的 ingestEndpoint 响应字段中接收的主机名的前面应为 `rtmps://`，后面为 `/app`。完整 URL 应采用以下格式：`rtmps://{ ingestEndpoint }/app`

```
try broadcastSession.start(with: IVS_RTMP_URL, streamKey: IVS_STREAMKEY)
```

iOS 广播 SDK 仅支持 RTMPS 采集（不支持不安全的 RTMP 采集）。

## 停止广播

```
broadcastSession.stop()
```

## 管理生命周期事件

### 音频中断

在多种情况下，广播开发工具包将不具有音频输入硬件的独占访问权限。您需要处理的一些示例情况如下：

- 用户接到电话或来 FaceTime 电
- 用户激活 Siri

Apple 可以轻松地响应这些事件，方法是订阅 `AVAudioSession.interruptionNotification`：

```
NotificationCenter.default.addObserver(
    self,
    selector: #selector(audioSessionInterrupted(_:)),
    name: AVAudioSession.interruptionNotification,
    object: nil)
```

然后您可以使用类似方法来处理事件：

```
// This assumes you have a variable `isRunning` which tracks if the broadcast is
// currently live, and another variable `wasRunningBeforeInterruption` which tracks
// whether the broadcast was active before this interruption to determine if it should
// resume after the interruption has ended.

@objc
private func audioSessionInterrupted(_ notification: Notification) {
    guard let userInfo = notification.userInfo,
          let typeValue = userInfo[AVAudioSessionInterruptionTypeKey] as? UInt,
          let type = AVAudioSession.InterruptionType(rawValue: typeValue)
    else {
        return
    }
    switch type {
    case .began:
        wasRunningBeforeInterruption = isRunning
        if isRunning {
            broadcastSession.stop()
        }
    case .ended:
        defer {
            wasRunningBeforeInterruption = false
        }
        guard let optionsValue = userInfo[AVAudioSessionInterruptionOptionKey] as? UInt
        else { return }
        let options = AVAudioSession.InterruptionOptions(rawValue: optionsValue)
        if options.contains(.shouldResume) && wasRunningBeforeInterruption {
            try broadcastSession.start(
                with: IVS_RTMP_URL,
                streamKey: IVS_STREAMKEY)
        }
    @unknown default: break
    }
}
```

## 应用程序进入后台

不允许 iOS 中的标准应用程序在后台使用摄像机。在后台进行视频编码也有限制：由于硬件编码器受到限制，因此只有前台应用程序才能访问。因此，广播开发工具包会自动终止其会话并将其 `isReady` 属性设置为 `false`。当您的应用程序即将再次进入前台时，广播开发工具包会将所有设备重新连接到其原始 `IVSMixerSlotConfiguration` 条目。

广播开发工具包通过响应 `UIApplication.didEnterBackgroundNotification` 和 `UIApplication.willEnterForegroundNotification` 完成此操作。

如果您正在提供自定义图像源，您应该准备好处理这些通知。在终止流之前，您可能需要执行额外的步骤来停用它们。

请参阅[使用后台视频](#)以获取在应用程序处于后台时启用串流的解决方法。

## 媒体服务丢失

在极少数情况下，iOS 设备上的整个媒体子系统将崩溃。在这种情况下，我们无法再广播。您的应用程序对这些通知作出适当响应。至少应订阅以下通知：

- [mediaServicesWereLostNotification](#)—通过停止广播并完全取消分配来 `IVSBroadcastSession` 做出回应。广播会话使用的所有内部组件都将失效。
- [mediaServicesWereResetNotification](#)—通过通知您的用户他们可以再次进行广播来回应。根据您的使用案例，您可在此时重新自动开始广播。

## 高级使用案例

我们将在此处介绍一些高级使用案例。从上面的基本设置开始，然后在此处继续。

### 创建广播配置

我们将在此处创建一个带有两个混合器插槽的自定义配置，允许我们将两个视频源绑定到混合器。一个 (`custom`) 是全屏幕并布置在另一个 (`camera`) 后面，它更小且位于右下角。请注意，对于 `custom` 插槽，我们不设置位置、大小或宽高比模式。因为我们不设置这些参数，所以插槽使用视频设置的大小和位置。

```
let config = IVSBroadcastConfiguration()
try config.audio.setBitrate(128_000)
try config.video.setMaxBitrate(3_500_000)
try config.video.setMinBitrate(500_000)
try config.video.setInitialBitrate(1_500_000)
try config.video.setSize(CGSize(width: 1280, height: 720))
config.video.defaultAspectMode = .fit
config.mixer.slots = [
    try {
        let slot = IVSMixerSlotConfiguration()
        // Do not automatically bind to a source
        slot.preferredAudioInput = .unknown
```

```

        // Bind to user image if unbound
        slot.preferredVideoInput = .userImage
        try slot.setName("custom")
        return slot
    }(),
    try {
        let slot = IVSMixerSlotConfiguration()
        slot.zIndex = 1
        slot.aspect = .fill
        slot.size = CGSize(width: 300, height: 300)
        slot.position = CGPoint(x: config.video.size.width - 400, y:
config.video.size.height - 400)
        try slot.setName("camera")
        return slot
    }()
]

```

## 创建广播会话 (高级版)

就像在[基本示例](#)中一样创建 `IVSBroadcastSession`，但在此处会提供您的自定义配置。还对设备阵列提供 `nil`，因为我们将手动添加它们。

```

let broadcastSession = try IVSBroadcastSession(
    configuration: config, // The configuration we created above
    descriptors: nil, // We'll manually attach devices after
    delegate: self)

```

## 迭代和连接摄像机设备

我们将通过开发工具包检测到的输入设备进行迭代。开发工具包仅返回 iOS 中的内置设备。即使连接了蓝牙音频设备，它们也会显示为内置设备。有关更多信息，请参阅[已知问题和解决方法](#)。

找到我们想要使用的设备之后，我们会调用 `attachDevice` 来连接它。

```

let frontCamera = IVSBroadcastSession.listAvailableDevices()
    .filter { $0.type == .camera && $0.position == .front }
    .first
if let camera = frontCamera {
    broadcastSession.attach(camera, toSlotWithName: "camera") { device, error in
        // check error
    }
}

```

## 交换摄像机

```
// This assumes you've kept a reference called `currentCamera` that points to the
current camera.
let wants: IVSDevicePosition = (currentCamera.descriptor().position
  == .front) ? .back : .front
// Remove the current preview view since the device will be changing.
previewView.subviews.forEach { $0.removeFromSuperview() }
let foundCamera = IVSBroadcastSession
  .listAvailableDevices()
  .first { $0.type == .camera && $0.position == wants }
guard let newCamera = foundCamera else { return }
broadcastSession.exchangeOldDevice(currentCamera, withNewDevice: newCamera)
{ newDevice, _ in
  currentCamera = newDevice
  if let camera = newDevice as? IVSImageDevice {
    do {
      previewView.addSubview(try finalCamera.previewView())
    } catch {
      print("Error creating preview view \(error)")
    }
  }
}
```

## 创建自定义输入源

若要输入应用程序生成的声音或图像数据，请使用 `createImageSource` 或者 `createAudioSource`。这两种方法都会创建虚拟设备（`IVSCustomImageSource` 和 `IVSCustomAudioSource`），这些设备可以像任何其他设备一样绑定到混合器。

这两种方法返回的设备通过其 `onSampleBuffer` 函数接受 `CMSampleBuffer`：

- 对于视频源，像素格式必须为 `kCVPixelFormatType_32BGRA`、`420YpCbCr8BiPlanarFullRange` 或者 `420YpCbCr8BiPlanarVideoRange`。
- 对于音频源，缓冲区必须包含线性 PCM 数据。

您无法一起使用 `AVCaptureSession` 与摄像机输入以馈送自定义图像源，同时还使用广播开发工具包提供的摄像机设备。如果要同时使用多个摄像机，请使用 `AVCaptureMultiCamSession` 并提供两个自定义图像源。

自定义图像源主要应与静态内容（如图像）或视频内容一起使用：

```
let customImageSource = broadcastSession.createImageSource(withName: "video")
try broadcastSession.attach(customImageSource, toSlotWithName: "custom")
```

## 监控网络连接

移动设备通常会在使用过程中暂时失去并重新获得网络连接。因此，监控应用的网络连接并在事情发生变化时作出适当响应非常重要。

当广播连接断开时，广播开发工具包的状态将更改为 `error`，然后变为 `disconnected`。将通过 `IVSBroadcastSessionDelegate` 通知您发生了这些更改。当您收到这些状态更改时：

1. 您的连接恢复之后，监控广播应用程序的连接状态，并使用端点和流密钥调用 `start`。
2. 重要提示：监控状态委托回调，并确保在再次调用 `start` 之后状态更改为 `connected`。

## 分离设备

如果要分离而不是替换设备，请使用 `IVSDevice` 或 `IVSDeviceDescriptor` 来分离它。

```
broadcastSession.detachDevice(currentCamera)
```

## ReplayKit 集成

要在 iOS 上流式传输设备的屏幕和系统音频，必须与集成 [ReplayKit](#)。Amazon IVS 广播软件开发工具包便于 `ReplayKit` 使用进行 `IVSReplayKitBroadcastSession` 集成。在您的 `RPBroadcastSampleHandler` 子类中，创建 `IVSReplayKitBroadcastSession` 的实例，然后：

- 在 `broadcastStarted` 中开启会话
- 在 `broadcastFinished` 中停止会话

会话对象将有三个用于屏幕图像、应用程序音频和麦克风音频的自定义源。将 `processSampleBuffer` 中提供的 `CMSampleBuffers` 传递到这些自定义源。

要处理设备方向，您需要从样本缓冲区中提取 `ReplayKit` 特定元数据。使用以下代码：

```
let imageSource = session.systemImageSource;
```

```
if let orientationAttachment = CMGetAttachment(sampleBuffer, key:
  RPVideoSampleOrientationKey as CFString, attachmentModeOut: nil) as? NSNumber,
  let orientation = CGImagePropertyOrientation(rawValue:
orientationAttachment.uint32Value) {
  switch orientation {
  case .up, .upMirrored:
    imageSource.setHandsetRotation(0)
  case .down, .downMirrored:
    imageSource.setHandsetRotation(Float.pi)
  case .right, .rightMirrored:
    imageSource.setHandsetRotation(-(Float.pi / 2))
  case .left, .leftMirrored:
    imageSource.setHandsetRotation((Float.pi / 2))
  }
}
```

可以使用 `IVSBroadcastSession` 代替 `ReplayKit` 进行集成 `IVSReplayKitBroadcastSession`。但是，`ReplayKit` 特定变体进行了一些修改，以减少内部存储器占用，以保持 Apple 的广播扩展内存上限之内。

## 获取推荐的广播设置

若要在开始广播之前评估用户的连接，请使用

`IVSBroadcastSession.recommendedVideoSettings` 来运行一个简短的测试。在测试运行时，您将收到多个建议，从最推荐到最不推荐的顺序排列。在此版本的开发工具包中，无法重新配置当前 `IVSBroadcastSession`，因此您必须取消分配它，然后使用推荐的设置创建一个新的。您将继续收到 `IVSBroadcastSessionTestResults`，直到 `result.status` 为 `Success` 或者 `Error`。您可以使用 `result.progress` 检查进度。

Amazon IVS 支持 8.5 Mbps 的最大比特率（对于其 `type` 为 `STANDARD` 或 `ADVANCED` 的通道），所以此方法返回的 `maximumBitrate` 永远不会超过 8.5 Mbps。考虑到网络性能的小波动，建议此方法返回的 `initialBitrate` 略低于测试中测量的真实比特率。（通常不建议使用 100% 的可用带宽。）

```
func runBroadcastTest() {
  self.test = session.recommendedVideoSettings(with: IVS_RTMP_URL, streamKey:
IVS_STREAMKEY) { [weak self] result in
    if result.status == .success {
      this.recommendation = result.recommendations[0];
    }
  }
}
```



```
}
```

## 使用后台视频

即使您的应用程序处于后台，您也可以继续进行非RelayKit 广播。

为了节省电力并保持前台应用程序的响应速度，iOS 一次只允许一个应用程序访问 GPU。Amazon IVS 广播开发工具包在视频管道的多个阶段使用 GPU，包括合成多个输入源、缩放图像和对图像进行编码。尽管广播应用程序在后台运行，但不能保证开发工具包能够执行任何这些操作。

使用 `createAppBackgroundImageSource` 方法解决此问题。它使开发工具包能够在后台继续广播视频和音频。它将返回 `IVSBackgroundImageSource`，这是一个正常的 `IVSCustomImageSource`，且具有额外的 `finish` 功能。每一个提供给背景图像源的 `CMSampleBuffer` 按照原始 `IVSVideoConfiguration` 提供的帧速率进行编码。`CMSampleBuffer` 上的时间戳将被忽略。

然后，开发工具包对这些图像进行缩放和编码并进行缓存，从而在应用程序进入后台时自动循环该源。当您的应用程序返回到前台时，连接的图像设备再次变为活动状态，并且预编码的流停止循环。

要撤消此过程，请使用 `removeImageSourceOnAppBackgrounded`。除非您想显式恢复开发工具包的后台行为，否则您不必调用它；它将在取消分配 `IVSBroadcastSession` 时以其他方式被自动清除。

注意：我们强烈建议您在会话上线之前调用此方法，作为配置广播会话的一部分。该方法很昂贵（它对视频进行编码），因此在运行此方法时直播的性能可能会降低。

示例：为后台视频生成静态图像

向背景源提供单张图像会生成该静态图像的完整 GOP。

下面是使用 `CIImage` 的示例：

```
// Create the background image source
guard let source = session.createAppBackgroundImageSource(withAttemptTrim: true,
  onComplete: { error in
    print("Background Video Generation Done - Error: \(error.debugDescription)")
  }) else {
  return
}

// Create a CIImage of the color red.
let ciImage = CIImage(color: .red)
```

```
// Convert the UIImage to a CVPixelBuffer
let attrs = [
    kCVPixelBufferCGImageCompatibilityKey: kCFBooleanTrue,
    kCVPixelBufferCGBitmapContextCompatibilityKey: kCFBooleanTrue,
    kCVPixelBufferMetalCompatibilityKey: kCFBooleanTrue,
] as CFDictionary

var pixelBuffer: CVPixelBuffer!
CVPixelBufferCreate(kCFAllocatorDefault,
    videoConfig.width,
    videoConfig.height,
    kCVPixelFormatType_420YpCbCr8BiPlanarFullRange,
    attrs,
    &pixelBuffer)

let context = CIContext()
context.render(ciImage, to: pixelBuffer)

// Submit to CVPixelBuffer and finish the source
source.add(pixelBuffer)
source.finish()
```

或者，您可以使用捆绑图像，而不是创建纯色 `UIImage`。此处显示的唯一代码是如何将 `UIImage` 转换为 `UIImage` 以便与上一个示例一起使用：

```
// Load the pre-bundled image and get it's CGImage
guard let cgImage = UIImage(named: "image")?.cgImage else {
    return
}

// Create a UIImage from the CGImage
let ciImage = UIImage(cgImage: cgImage)
```

示例：带有 AV 的视频 `AssetImageGenerator`

您可以使用 `AVAssetImageGenerator` 生成来自 `AVAsset` 的 `CMSampleBuffers`（尽管不是通过 HLS 流 `AVAsset`）：

```
// Create the background image source
guard let source = session.createAppBackgroundImageSource(withAttemptTrim: true,
    onComplete: { error in
        print("Background Video Generation Done - Error: \(error.debugDescription)")
    }) else {
```

```
    return
}

// Find the URL for the pre-bundled MP4 file
guard let url = Bundle.main.url(forResource: "sample-clip", withExtension: "mp4") else
{
    return
}
// Create an image generator from an asset created from the URL.
let generator = AVAssetImageGenerator(asset: AVAsset(url: url))
// It is important to specify a very small time tolerance.
generator.requestedTimeToleranceAfter = .zero
generator.requestedTimeToleranceBefore = .zero

// At 30 fps, this will generate 4 seconds worth of samples.
let times: [NSValue] = (0...120).map { NSValue(time: CMTime(value: $0, timescale:
    CMTimeScale(config.video.targetFramerate))) }
var completed = 0

let context = CIContext(options: [.workingColorSpace: NSNull()])

// Create a pixel buffer pool to efficiently feed the source
let attrs = [
    kCVPixelBufferPixelFormatTypeKey: kCVPixelFormatType_420YpCbCr8BiPlanarFullRange,
    kCVPixelBufferCGImageCompatibilityKey: kCFBooleanTrue,
    kCVPixelBufferCGBitmapContextCompatibilityKey: kCFBooleanTrue,
    kCVPixelBufferMetalCompatibilityKey: kCFBooleanTrue,
    kCVPixelBufferWidthKey: videoConfig.width,
    kCVPixelBufferHeightKey: videoConfig.height,
] as CFDictionary
var pool: CVPixelBufferPool!
CVPixelBufferPoolCreate(kCFAllocatorDefault, nil, attrs, &pool)

generator.generateCGImagesAsynchronously(forTimes: times) { requestTime, image,
    actualTime, result, error in
    if let image = image {
        // convert to CIImage then CVpixelBuffer
        let ciImage = CIImage(cgImage: image)
        var pixelBuffer: CVPixelBuffer!
        CVPixelBufferPoolCreatePixelBuffer(kCFAllocatorDefault, pool, &pixelBuffer)
        context.render(ciImage, to: pixelBuffer)
        source.add(pixelBuffer)
    }
    completed += 1
```

```
if completed == times.count {
    // Mark the source finished when all images have been processed
    source.finish()
}
}
```

可以使用 `AVPlayer` 和 `AVPlayerItemVideoOutput` 生成 `CVPixelBuffers`。但是，这需要使用 `CADisplayLink` 并且更接近实时地执行，而 `AVAssetImageGenerator` 可以更快地处理帧。

## 限制

您的应用程序需要[后台音频授权](#)以避免在进入后台后被暂停。

`createAppBackgroundImageSource` 只能在应用程序处于前台时被调用，因为它需要访问 GPU 才能完成。

`createAppBackgroundImageSource` 始终编码为完整的 GOP。例如，如果您的关键帧间隔为 2 秒（默认值）且运行速度为 30 fps，则会对 60 帧的倍数进行编码。

- 如果提供的帧少于 60 帧，则无论去除选项的值如何，都会重复最后一帧，直到达到 60 帧。
- 如果提供的帧数超过 60 帧并且去除选项为 `true`，则最后 N 帧将被丢弃，其中 N 等于提交帧总数的剩余部分除以 60。
- 如果提供的帧数超过 60 帧并且去除选项为 `false`，将重复最后一帧，直到达到 60 帧的下一个倍数。

## iOS 如何选择相机分辨率和帧率

由广播 SDK 管理的摄像机可优化其分辨率和帧速率（或 FPS）frames-per-second，以最大限度地减少热量产生和能耗。本节介绍如何选择分辨率和帧率以帮助主机应用程序针对其使用案例进行优化。

将 `IVSCamera` 连接到 `IVSBroadcastSession` 时，会根据帧率 `IVSVideoConfiguration.targetFramerate` 和分辨率 `IVSVideoConfiguration.size` 优化相机。这些值是在初始化时提供给 `IVSBroadcastSession` 的。

## 已知问题和解决方法

- 在直播期间插入有线头戴式耳机时，错误 `ReplayKit` 会导致内存快速增长。

解决办法：使用已插入的有线耳机来开启流，使用蓝牙耳机或不使用外接麦克风。

- 如果您在 ReplayKit 直播期间的任何时候启用麦克风然后中断音频会话（例如，通过电话或激活 Siri），则系统音频将停止工作。这是我们正在与 Apple 合作解决的 ReplayKit 错误。

解决办法：在音频中断时，停止广播并向用户发出警报。

- AirPods 如果 AVAudioSession 类别设置为，则不要录制任何音频 record。默认情况下，开发工具包使用 playAndRecord，所以此问题仅在类别更改为 record 时出现。

解决办法：如果有可能使用它来录制音频，playAndRecord 即使您的应用程序没有播放媒体，也要使用。AirPods

- 连接 AirPods 到 iOS 12 设备时，不能使用其他麦克风录制音频。尝试切换到内置麦克风会立即恢复为。AirPods

解决办法：尚无。如果 AirPods 连接到 iOS 12，则它们是唯一可以录制音频的设备。

- 提交音频数据的速度比实时更快（使用自定义音频源）会导致音频偏差。

解决办法：不要以快于实时的速度提交音频数据。

- 使用高采样率（44100 Hz 或更高）和两个通道时，音频构件可能以低于 68 kbps 的比特率出现。

解决办法：将比特率提高到 68 kbps 或更高，将采样率降至 24000 Hz 或更低，或将通道设置为 1。

- 在 IVSMicrophone 设备上启用回声消除时，listAvailableInputSources 方法仅返回单个麦克风源。

解决办法：尚无。此行为由 iOS 控制。

- 更改蓝牙音频路由是不可预测的。如果您在会话中连接新设备，iOS 可能会自动更改输入路由，也可能不会。此外，无法在同一时间连接的多个蓝牙耳机之间进行选择。常规广播和舞台会话中均会发生此现象。

解决办法：如果您打算使用蓝牙耳机，请在开始广播或进入舞台之前进行连接，并在整个会话期间保持连接状态。

- iOS 在打开配对的 AirPods 手机壳后出现 AirPods 弹出窗口时会移除对相机的访问权限，同时将 AirPods 手机置于保护套中。这会导致用于广播或舞台的视频冻结。

解决办法：无。在弹出窗口呈现过程中，iOS 会完全撤消对相机的访问权限，第三方应用程序无法阻止弹出窗口。

## IVS Broadcast SDK: Mixer Guide (低延迟流式传输)

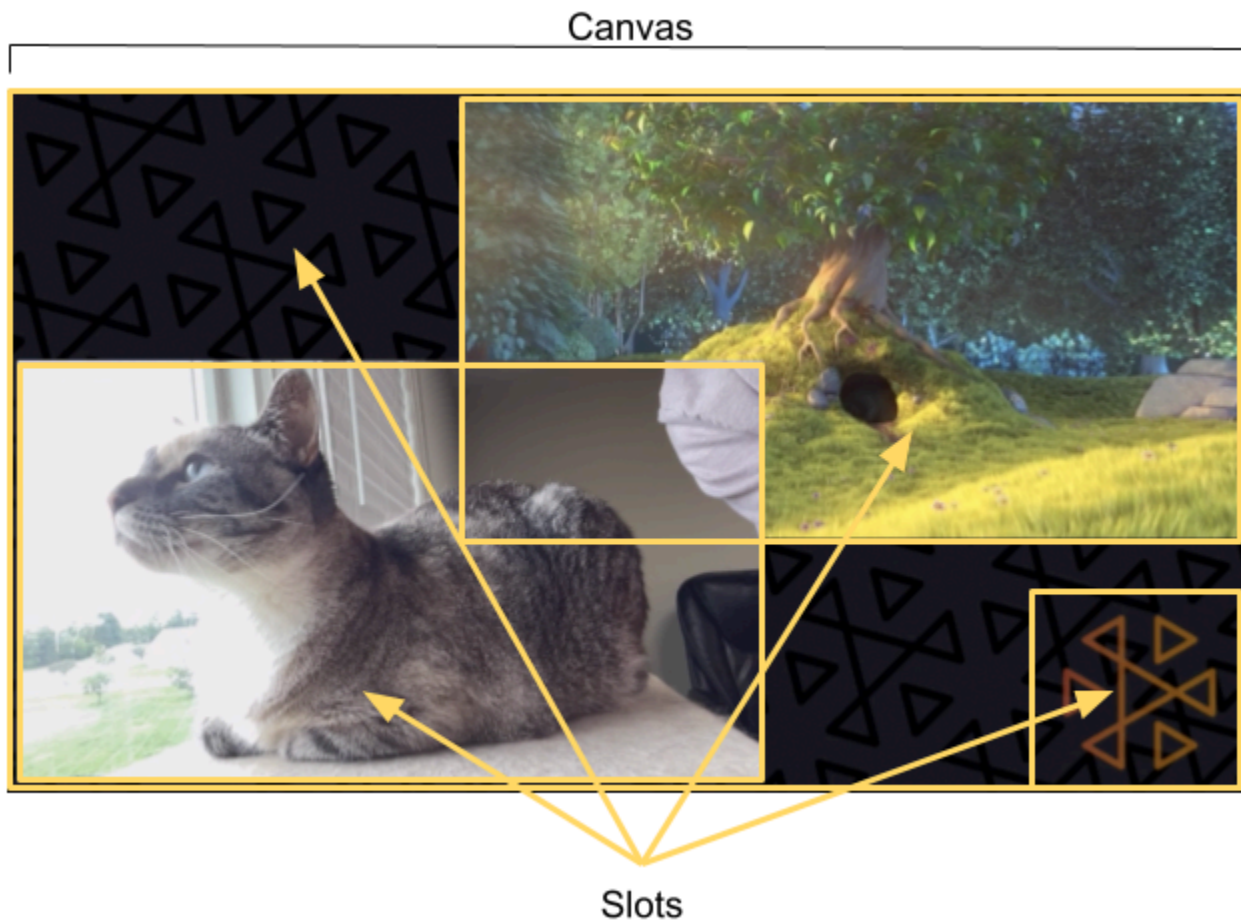
混合器是一种音频和视频处理单元，它接收多个输入源并生成单个输出。它是一项强大的功能，可让您定义和管理多个屏幕上的（视频）元素和音轨。您可以组合来自多个来源的视频和音频，例如摄像头、麦克风、屏幕截图以及应用程序生成的音频和视频。您可以使用转换围绕流式传输到 Amazon IVS 的视频移动这些源，然后在流中添加和删除它们。

要访问混合器，请致电：

Android 上的 `BroadcastSession.getMixer()`

iOS 上的 `IVSBroadcastSession.mixer`

### 术语



租期	描述
Binding	要将输入设备与插槽关联，该设备必须绑定到混合器插槽。该操作通过 <code>Mixer.bind()</code> 方法完成。一个插槽一次可以绑定一个图像输入和一个音频输入。您可以通过调用 <code>Mixer.unbind()</code> 来解除设备与插槽的绑定。
画布	在您的 <code>BroadcastSession</code> 配置中定义的视频的显示范围。画布的大小与您的视频设置相等，并以配置中指定的相同帧速率运行。
设备	一种硬件或软件组件，用于生成到 <code>BroadcastSession</code> 的音频或图像输入。例如，设备包括麦克风、摄像头、蓝牙耳机和屏幕截图或自定义图像输入等虚拟设备。除了自定义输入之外，通常不需要保留对设备对象的引用；相反，您可以保留设备描述符的副本。
设备描述符	包含有关输入设备的信息的结构，例如，其类型、系统地址、人类可读的“友好”名称以及移动设备上的物理位置。通过此信息，您可以决定是否要使用引用的设备并允许 Amazon IVS 访问该设备。
Slot	<p>定义视觉元素在屏幕上的位置以及音频混合中音轨属性的容器。混合器可以配置零个或多个插槽。系统将为插槽指定一个字符串名称，可用于绑定设备和执行转换。上图显示了四个插槽：</p> <ul style="list-style-type: none"> <li>• 左下带摄像头输入</li> <li>• 右上带电影输入</li> <li>• 右下带有 Amazon IVS 徽标</li> <li>• 全屏背景图像</li> </ul> <p>配置会话后，您可以使用 <code>addSlot</code> 和 <code>removeSlot</code> 混合器方法添加和删除插槽。</p>
Transition	<p>要将插槽移动到新位置或更改其部分属性，请使用 <code>Mixer.transition()</code> 。此方法需要：</p> <ul style="list-style-type: none"> <li>• 代表插槽下一个状态的新插槽结构</li> <li>• 一个持续时间，该持续时间指定动画相对于视频的时间轴应该花费多长时间。如果持续时间设置为 0，则转换发生在混合的下一帧上。</li> <li>• 一个可选的回调函数，通知您动画的完成时间。回调可能对于链接动画很有用。</li> </ul>



## 画布属性

画布属性基于您在创建 `BroadcastSession` 时提供的 `BroadcastConfiguration` 设置。Audio 和 Video 结构中的几个属性会影响画布：

名称	类型	描述
<code>Audio.channels</code>	整数	音频混合器的输出通道数。有效值：1、2。1 通道为单声道音频；2 为立体声音频。默认值：2。
<code>Audio.sampleRate</code>	<code>AudioSampleRate</code>	音频混合器每秒发出的音频样本数。该值应至少是音频信号中最高频率的两倍。人们最高可以听到约 20 kHz 的声音，因此 44.1 kHz 和 48 kHz 一般就足够了。默认值：48 kHz。
<code>Video.defaultAspectMode</code>	<code>AspectMode</code>	插槽的默认宽高比模式。有效值： <ul style="list-style-type: none"> <li>• <code>Fill</code> — 保持图像的宽高比但填满插槽。必要时，图像将被裁剪。</li> <li>• <code>Fit</code> — 保持图像的宽高比但将整个图像放入插槽中。必要时，插槽可能有一个宽银幕效果或左右黑边。如果设置了该值，宽银幕效果/左右黑边将为 <code>fillColor</code>，否则，将为透明（如果图像后的画布颜色为黑色，它看起来可能是黑色）。</li> <li>• <code>None</code> — 请勿保持图像的宽高比。图像将被缩放，以匹配插槽的尺寸。</li> </ul>
<code>Video.size</code>	<code>Vec2</code>	视频画布的大小。
<code>Video.targetFrameRate</code>	整数	画布的每秒目标帧数。通常情况下，该值应能够得到满足，但是在某些情况下（例如 CPU 负载过高或网络拥塞），系统可能会丢弃帧。

## 插槽属性

插槽有几个可配置属性，您可以使用这些属性来自定义场景和制作动画。对于持续时间超过 0 秒的转换，任何为 `Float` 或 `Vector` 的值都将被使用线性插值制作动画。



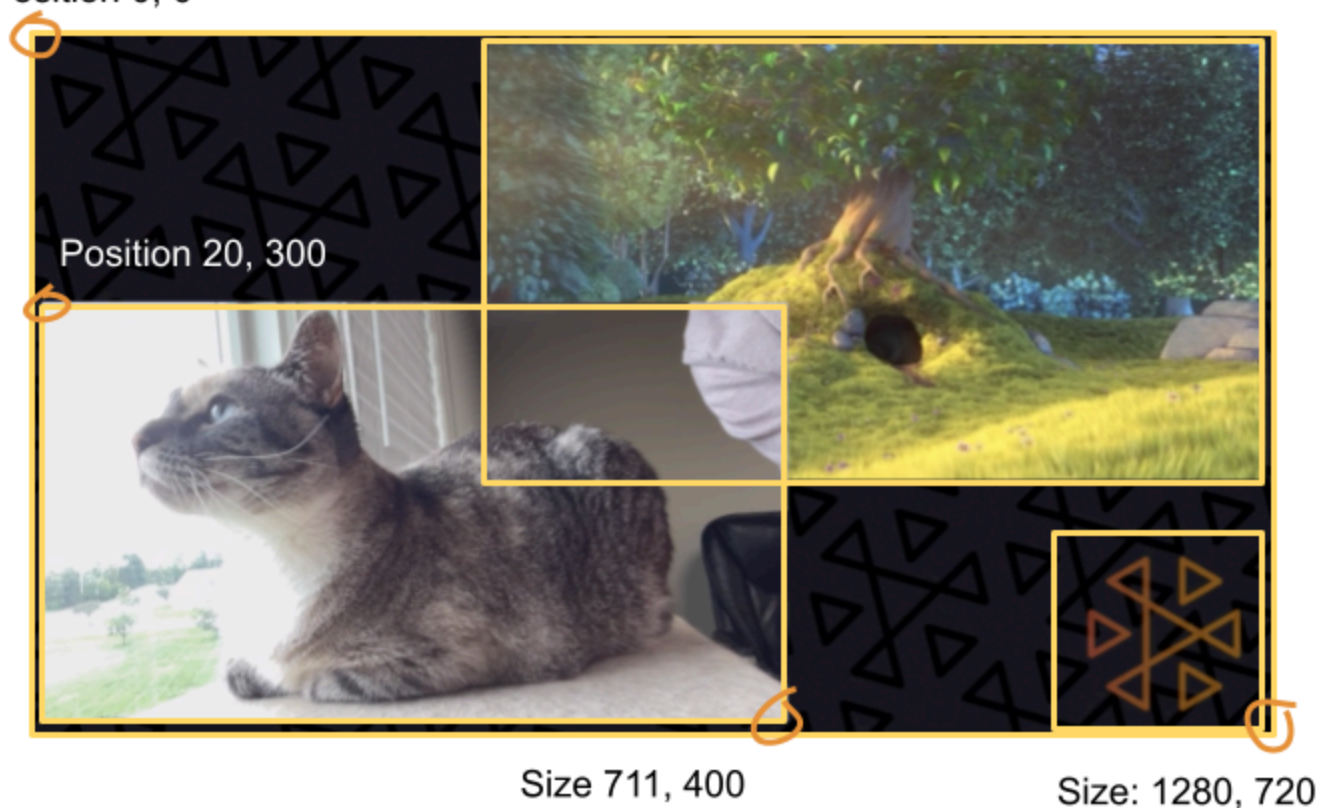
名称	类型	描述
aspect	AspectMode	<p>适用于插槽中渲染的任何图像的宽高比模式。有效值：</p> <ul style="list-style-type: none"> <li>• Fill — 保持图像的宽高比但填满插槽。必要时，图像将被裁剪。</li> <li>• Fit — 保持图像的宽高比但将整个图像放入插槽中。必要时，插槽可能有一个宽银幕效果或左右黑边。如果设置了该值，宽银幕效果/左右黑边将为 fillColor ，否则，将为透明（如果图像后的画布颜色为黑色，它看起来可能是黑色）。</li> <li>• None — 请勿保持图像的宽高比。图像将被缩放，以匹配插槽的尺寸。</li> </ul> <p>默认值：如果 matchCanvasAspectMode 为真，与画布 aspect 相同，否则为 Fill。设置此值也会将 matchCanvasAspectMode 设置为假。</p>
fillColor	Vec4	<p>当插槽和图像的宽高比不匹配时，填充用于 Aspect Fit 的颜色。格式为 ( red、green、blue、alpha )。有效值（对于每个通道）：0-1。默认值：(0, 0, 0, 0)。</p>
gain	Float	<p>音频增益。这是一个乘数，因此任何高于 1 的值都会提高增益；任何低于 1 的值都会减小增益。有效值：0 - 2。默认值：1。</p>
matchCanvasAspectMode	布尔值	<p>如果为真，请使用画布的 Video.defaultAspectMode 值。如果您设置插槽的 aspect 属性，则该值被设置为假。默认值：真。</p>
matchCanvasSize	布尔值	<p>如果为真，则插槽的大小调整为等于画布的大小，其位置设置为 (0, 0)。如果您设置插槽的 size 属性，则该值被设置为假。默认值：真。</p>
name	字符串	<p>插槽的名称。该名称用于引用插槽以进行绑定和转换。默认值: "default" 。</p>

名称	类型	描述
position	Vec2	插槽相对于画布左上角的位置 ( 以像素为单位 )。插槽的原点也在左上角。
preferredAudioInput	DeviceType	<p>首选的音频输入设备类型。如果解除此插槽的绑定并将指定类型的音频设备连接到会话，则该设备将自动绑定到此插槽。有效值：</p> <ul style="list-style-type: none"> <li>• 麦克风 — 音频硬件，例如板载麦克风、可插拔耳机或蓝牙耳机。</li> <li>• 系统音频 — 从操作系统捕获的音频，通常伴随屏幕录制。</li> <li>• 用户音频 — 您创建的自定义音频输入。</li> <li>• 未知 — 没有首选设备；插槽将始终需要手动绑定。</li> </ul>
preferredVideoInput	DeviceType	<p>首选的视频输入设备。如果解除此插槽的绑定并将指定类型的视频设备连接到会话，则该设备将自动绑定到此插槽。有效值：</p> <ul style="list-style-type: none"> <li>• 摄像头 — 板载摄像头设备，例如前置、后置或广角摄像头。</li> <li>• 屏幕 — 来自操作系统的屏幕截图。</li> <li>• 用户图像 — 您创建的自定义图像和视频输入。</li> <li>• 未知 — 没有首选设备；插槽将始终需要手动绑定。</li> </ul>
size	Vec2	插槽的大小 ( 以像素为单位 )。设置此值也会将 matchCanvasSize 设置为假。默认值：(0, 0)；但是，因为 matchCanvasSize 默认为真，插槽的渲染大小就是画布大小，而不是 (0, 0)。
transparency	Float	插槽的透明度。这是与图像中的任何 Alpha 值相乘的结果。不透明度为 1 - transparency。有效值：0-1，其中 0 表示完全不透明，1 表示完全透明。默认值：0。

名称	类型	描述
zIndex	Float	插槽的相对顺序。zIndex 值较高的插槽绘制在 zIndex 值较低的插槽之上。

## 配置广播会话以进行混合

Position 0, 0



在这里，我们创建了一个类似于本指南开头的场景，其中包含三个屏幕元素：

- 左下角的摄像头插槽。
- 右下角的徽标叠加插槽。
- 右上角用于电影的插槽。

请注意，画布的原点是左上角，插槽也是一样。因此，将插槽定位在 (0, 0) 会将其放在左上角，且整个插槽可见。

## iOS

```
let config = IVSBroadcastConfiguration()
try config.video.setSize(CGSize(width: 1280, height: 720))
try config.video.setTargetFramerate(60)
config.video.enableTransparency = true

// Bottom Left
var cameraSlot = IVSMixerSlotConfiguration()
cameraSlot.size = CGSize(width: 320, height: 180)
cameraSlot.position = CGPoint(x: 20, y: 1280 - 200)
cameraSlot.preferredVideoInput = .camera
cameraSlot.preferredAudioInput = .microphone
cameraSlot.matchCanvasAspectMode = false
cameraSlot.zIndex = 2
try cameraSlot.setName("camera")

// Top Right
var streamSlot = IVSMixerSlotConfiguration()
streamSlot.size = CGSize(width: 640, height: 320)
streamSlot.position = CGPoint(x: 1280 - 660, y: 20)
streamSlot.preferredVideoInput = .userImage
streamSlot.preferredAudioInput = .userAudio
streamSlot.matchCanvasAspectMode = false
streamSlot.zIndex = 1
try streamSlot.setName("stream")

// Bottom Right
var logoSlot = IVSMixerSlotConfiguration()
logoSlot.size = CGSize(width: 320, height: 180)
logoSlot.position = CGPoint(x: 1280 - 340, y: 720 - 200)
logoSlot.preferredVideoInput = .userImage
logoSlot.preferredAudioInput = .unknown
logoSlot.matchCanvasAspectMode = false
logoSlot.zIndex = 3
try logoSlot.setTransparency(0.7)
try logoSlot.setName("logo")

config.mixer.slots = [ cameraSlot, streamSlot, logoSlot ]
```

## Android

```
// Bottom Left
```

```
val cameraSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(320, 180)
    s.position = BroadcastConfiguration.Vec2(20, 1280 - 200)
    s.preferredVideoInput = Device.Descriptor.DeviceType.CAMERA
    s.preferredAudioInput = Device.Descriptor.DeviceType.MICROPHONE
    s.matchCanvasAspectMode = false
    s.zIndex = 2
    s.name = "camera"
    s
}

// Top Right
val streamSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(640, 320)
    s.position = BroadcastConfiguration.Vec2(1280 - 660, 20)
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.USER_AUDIO
    s.matchCanvasAspectMode = false
    s.zIndex = 1
    s.name = "stream"
    s
}

// Bottom Right
val logoSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(320, 180)
    s.position = BroadcastConfiguration.Vec2(1280 - 340, 720 - 200)
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.UNKNOWN
    s.matchCanvasAspectMode = false
    s.zIndex = 3
    s.name = "logo"
    s.transparency = 0.7
    s
}

val config = BroadcastConfiguration.with { c ->
    c.mixer.slots = listOf(cameraSlot, streamSlot, logoSlot)
    c.video.targetFramerate = 60
    c.video.setSize(1280, 720)
    c
}
```

## 添加槽

一旦您使用配置创建了 `BroadcastSession`，您可以在混合器中添加插槽和从中移除插槽。在这里，我们在混合器中添加了一个大的背景插槽用于图像。

### iOS

```
// Background. We will use most of the defaults for this slot.
var backgroundSlot = IVSMixerSlotConfiguration()
backgroundSlot.preferredVideoInput = .userImage
backgroundSlot.preferredAudioInput = .unknown
backgroundSlot.matchCanvasAspectMode = false
try backgroundSlot.setName("background")

session.mixer.addSlot(backgroundSlot)
```

### Android

```
// Background. We will use most of the defaults for this slot.
val backgroundSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.UNKNOWN
    s.matchCanvasAspectMode = false
    s.name = "background"
    s
}

session.mixer.addSlot(backgroundSlot)
```

## 移除插槽

要移除插槽，请使用您想要移除的插槽的名称调用 `BroadcastSession.Mixer.removeSlot`。绑定到该插槽的任何设备都会自动解除绑定，因此，您如果要继续使用它们，必须将它们重新绑定到不同的插槽。

## 动画转换

混合器转换方法用新配置替换插槽的配置。通过将持续时间设置为大于 0（以秒为单位），可以将此替换制作为随时间推移的动画。

## 哪些属性可以制作成动画？

插槽结构中并非所有属性都可以制作成动画。基于 Float 类型的任何属性都可以制作成动画；其他属性在动画的开始或结束时生效。

名称	可以制作成动画吗？	影响点
aspect	否	End
fillColor	是	已插入
gain	是	已插入
matchCanvasAspectMode	否	Start
matchCanvasSize	否	Start
name 注意：您不能更改插槽的名称。	否	不适用
position	是	已插入
preferredAudioInput	否	End
preferredVideoInput	否	End
size	是	已插入
transparency	是	已插入
zIndex 注意：zIndex 在 3D 空间中移动 2D 平面，因此当两个平面在动画中间的某个点交叉时，就会发生转换。这可以计算出来，但具体取决于开始和结束 zIndex 值。为了实现更顺畅的转换，请将其与 transparency 结合使用。	是	Unknown

## 简单示例

以下是使用上述[配置广播会话进行混合](#)中定义的配置进行全屏摄像头接管的示例。将其在 0.5 秒以内的变化制作成动画。

### iOS

```
// Bottom Left
var bigCameraSlot = cameraSlot
bigCameraSlot.size = CGSize(width: 1280, height: 720)
bigCameraSlot.position = CGPoint(x: 0, y: 0)

session.mixer.transition("camera", bigCameraSlot, 0.5) {
    println("animation completed!")
}
```

### Android

```
// Bottom Left
val bigCameraSlot = cameraSlot.changing { s ->
    s.setSize(1280, 720)
    s.position = BroadcastConfiguration.Vec2(0, 0)
    s
}

session.mixer.transition("camera", bigCameraSlot, 0.5) {
    print("animation completed!")
}
```

## 镜像广播

要在此方向镜像广播中附加的映像设备...	使用负值表示...
水平	槽的宽度
垂直	槽的高度
水平和垂直方向	槽的宽度和高度

需要按相同的值调整位置，以便在镜像时将槽置于正确的位置。



以下是水平和垂直镜像广播的示例。

## iOS

水平镜像：

```
var cameraSlot = IVSMixerSlotConfiguration
cameraSlot.size = CGSize(width: -320, height: 720)
// Add 320 to position x since our width is -320
cameraSlot.position = CGPoint(x: 320, y: 0)
```

垂直镜像：

```
var cameraSlot = IVSMixerSlotConfiguration
cameraSlot.size = CGSize(width: 320, height: -720)
// Add 720 to position y since our height is -720
cameraSlot.position = CGPoint(x: 0, y: 720)
```

## Android

水平镜像：

```
cameraSlot = BroadcastConfiguration.Mixer.Slot.with {
    it.size = BroadcastConfiguration.Vec2(-320f, 180f)
    // Add 320f to position x since our width is -320f
    it.position = BroadcastConfiguration.Vec2(320f, 0f)
    return@with it
}
```

垂直镜像：

```
cameraSlot = BroadcastConfiguration.Mixer.Slot.with {
    it.size = BroadcastConfiguration.Vec2(320f, -180f)
    // Add 180f to position y since our height is -180f
    it.position = BroadcastConfiguration.Vec2(0f, 180f)
    return@with it
}
```

注意：此镜像与 `ImagePreviewView` (Android) 和 `IVSImagePreviewView` (iOS) 上的 `setMirrored` 方法不同。该方法仅影响设备上的本地预览视图，不会影响广播。

## IVS 广播 SDK：自定义图像源（低延迟流式传输）

本指南假设您已熟悉如何设置广播会话（[Android](#)、[iOS](#)）以及如何[使用混合器 API](#)。

自定义图像输入源允许应用程序向广播 SDK 提供自己的图像输入，而不仅限于预设相机或屏幕共享。自定义图像源可以是简单的半透明水印或静态“马上回来”等场景，也可以允许应用程序执行额外的自定义处理，例如向相机添加美颜滤镜。

您可以拥有多个自定义图像源，例如水印加上带美颜滤镜的相机。当您使用自定义图像输入源对摄像机进行自定义控制时（例如使用需要访问相机的美颜滤镜库）时，广播 SDK 不再负责管理相机。相反，应用程序负责正确处理相机的生命周期。请参阅官方平台文档，以了解应用程序应如何管理相机。

### Android

创建广播会话后，请创建一个图像输入源：

```
SurfaceSource surfaceSource = broadcastSession.createImageInputSource();
```

此方法将返回一个 `SurfaceSource`，这是标准的 Android [Surface](#) 支持的图像源。它会自动附加到广播会话，因此之后无需使用 `attachDevice(...)` 方法。但是，`SurfaceSource` 需要绑定到一个插槽；这将在下文详细介绍。`SurfaceSource` 支持大小调整和轮换。您还可以创建 `ImagePreviewView` 来显示其内容预览。

检索底层 `Surface`：

```
Surface surface = surfaceSource.getInputSurface();
```

此 `Surface` 可以用作图像创建器（例如 `Camera2`、`OpenGL ES` 和其他库）的输出缓冲区。最简单的使用场景是直接在 `Surface` 画布中绘制静态位图或颜色。但是，许多库（例如 `beauty-filter` 库）提供了一种方法，允许应用程序指定外部 `Surface` 来进行渲染。您可以用这种方法将此 `Surface` 传递到滤镜库，从而让库输出处理后的帧以便广播会话进行流式传输。

最后，`SurfaceSource` 必须绑定到将由广播会话流式传输的 `Mixer.Slot`：

```
broadcastSession.getMixer().bind(surfaceSource, "customSlot");
```

此 [Android 示例代码](#) 包含了多个以不同方式使用自定义图像源的示例：

- 在 `MixerActivity` 中添加了一个半透明水印。

- 一个 MP4 文件在 MixerActivity 中回环。
- [CameraManager](#) 实用程序类使用 CustomActivity 中的 Camera2 方法对设备相机进行自定义管理，这种方法使用了一个简单的 sepia 滤镜。此示例特别有用，因为它演示了如何管理相机以及将广播会话的自定义 SurfaceSource 传递到相机捕获请求。如果您使用其他外部库，请参阅其文档以了解如何配置库，以输出到由广播会话提供的 Android Surface。

## iOS

创建广播会话后，请创建一个图像输入源：

```
let customSource = broadcastSession.createImageSource(withName: "customSourceName")
```

此方法将返回一个 IVSCustomImageSource，这是一个允许应用程序手动提交 CMSampleBuffers 的图像源。有关支持的像素格式，请参阅 iOS 广播 SDK 参考；指向最新版本的链接详见最新广播 SDK 发行版的 [Amazon IVS 版本注释](#)。源不会自动附加到广播会话，因此必须将图像源附加到会话并将其绑定到插槽，然后才能流式传输源：

```
broadcastSession.attach(customSource, toSlotWithName: "customSourceSlot", onComplete: nil)
```

附加并绑定自定义源后，应用程序可以直接将 CMSampleBuffers 提交到自定义源。您可以选择使用 onComplete 回调来开始操作。

提交到自定义源的样本将在广播会话中流式传输：

```
customSource.onSampleBuffer(sampleBuffer)
```

对于串流视频，请在回调中使用此方法。例如，假设您使用的是相机，则每次从 AVCaptureSession 收到一个新的样本缓冲时，应用程序可以将样本缓冲转发到自定义图像源。如果需要，应用程序可以在将样本提交到自定义图像源之前执行进一步的处理（例如美颜滤镜）。

对于静态图像，在收到第一个样本之后，如果自定义图像源的插槽绑定已更改，或源已分离并重新连接到广播会话，则应用程序需要重新提交样本。例如，假设删除了混合器的插槽然后将该插槽添加到混合器，则必须重新提交样本。

此 [iOS 示例应用程序](#) 包含了多个以不同方式使用自定义图像源的示例：

- 在 MixerViewController 中添加了一个半透明水印。

- 一个 MP4 文件在 `MixerViewController` 中回环。
- 在 `CustomSourcesViewController` 中添加了一个带有设备摄像头的 `CiFilter` 实现。这使应用程序能够独立于 Amazon IVS 广播开发工具包管理设备摄像头。它通过 `AVCaptureSession` 从设备摄像头捕获图像、使用 `CiFilter` 实现处理图像，然后将 `CMSampleBuffers` 提交到 `customSource` 来实现实时流。

# Amazon IVS 播放器开发工具包

要使用 Amazon Interactive Video Service (IVS)，您必须使用 Amazon IVS Player 播放器。该播放器是一个跨平台的软件开发工具包套件，用于播放 Amazon IVS 视频流。旨在利用 Amazon IVS 基础设施，并针对 Amazon IVS 播放进行了优化。

我们唯一可以保证性能的播放器是 Amazon IVS 播放器。为实现低延迟，需使用 Amazon IVS 播放器。

Amazon IVS 播放器的主要功能包括：

- **Ultra-low-latency streaming** — 低延迟是打造良好的交互式用户体验以丰富观众体验的关键组成部分。在播主和观众之间的传输路径中，延迟会逐渐增加，从而降低响应能力。  
  
End-to-end 延迟是指从摄像机上捕捉直播到直播出现在观看者屏幕上的延迟。Amazon IVS 旨在提供超低 end-to-end 延迟（不到五秒，具体取决于广播地点和广播公司设置）。为了实现这种低延迟，需使用 Amazon IVS 播放器。
- **跨平台一致性** – 观众可在各种平台上观看广播。从移动设备到 Web 浏览器，Amazon IVS 播放器都为所有观众提供了类似的体验。每个平台都使用相同的播放器函数库，所以这种一致性是可能的。播放器库是 Amazon IVS 基础设施不可或缺的组成部分。使用一个视频堆栈可确保所有视频播放行为（包括低延迟模式、定时元数据、分析、错误跟踪、报告和日志记录）在所有受支持的平台上都能以一致的方式提供。
- **自适应比特率流式处理 (ABR)** – Amazon IVS 播放器使用针对低延迟环境优化的 ABR 算法。播放器实时测量服务质量和带宽可用性，并调整视频质量和缓冲区级别，以提供不间断的播放。连接质量受到影响时，ABR 切换到较低的比特率；连接质量提高时，则会切换到较高的比特率。
- **定时元数据** – Amazon IVS 播放器支持定时元数据，可用于构建交互式元素，如投票和测验。元数据是描述并提供有关其他数据的信息的一组数据。使用“定时”元数据，时间码会随附视频流的数据片段。在播放过程中，时间码作为基于数据触发操作的提示点，例如：
  - 发送体育赛事的球员统计数据
  - 发送商品详情以进行实时购物
  - 发送问题以进行实时测验
- **强大的错误处理能力** – 很好地处理瞬态错误可避免观看体验中断。Amazon IVS Player 强大的错误处理能力可测许多潜在的流传输错误，自动切换到替代格式副本。观众可以不间断观看广播，而无需采取任何纠正措施。

- 易于集成 – Amazon IVS 播放器 API 缩小了 Amazon IVS 客户应用程序和播放器库之间的差距。API 对所有支持的平台都有绑定，使用熟悉的编码环境和技术，播放器可以轻松地集成到应用程序中。通过完全控制 UI 元素，客户可以自定义其应用程序的品牌和呈现方式。

Amazon IVS 播放器不支持使用 Airplay 进行转换。使用默认的 Chromecast 接收器应用程序，可以在播放器外部使用 Chromecast 转换。但是，这些应用程序中的延迟要比 Amazon IVS 播放器开发工具包中的延迟高，因此转换不会是无缝的。另请参阅适用于[低延迟直播功能](#)和适用于[实时直播功能](#)的关于 Amazon IVS 广播 SDK 的文档。

## 浏览器和平台要求

有关各种浏览器的最新发布版本的详细信息，请参阅：

- [Chrome 平台状态](#)
- [Firefox 版本](#)
- [Microsoft Edge 发布计划](#)
- [Safari 发布说明](#)

虽然 Amazon IVS 可能适用于某些旧版浏览器，但我们不会修复与旧版浏览器相关的错误。

类似浏览器的环境不支持 IVS 回放器 Web SDK (包括 Video.js 和 Player JW 集成)。这包括支持网络 WebViews 应用程序的原生设备和“10 英尺长的设备”(电视、游戏机、机顶盒)。如果您不确定下表之外的特定浏览器支持，请联系 IVS 技术支持。

## 桌面浏览器

桌面浏览器	支持的平台	受支持的版本
Chrome	Windows、macOS	两个主要版本 (当前版本和最新版本)
Firefox	Windows、macOS	两个主要版本 (当前版本和最新版本)
边缘	(Windows 8.1 和更高版本)	44.0 和更高版本  (在 <a href="#">Microsoft Edge Legacy</a> 自动质量模式下，仅支持正常延迟播放，而不支持超低延迟播放。自动质量模式是指是否已启用 ABR。例

桌面浏览器	支持的平台	受支持的版本
		如，在 Web 播放器上，请参阅 <code>setAutoQualityMode</code> 。
Safari	macOS	两个主要版本（当前版本和最新版本）  （在 Safari for macOS 14 及更高版本的自动质量模式下，IVS 播放器 1.3.0 及以上版本支持超低延迟播放。对于 Safari 浏览器和 IVS 播放器的早期版本，仅支持正常延迟播放。请参阅上面的“自动质量模式”。）

## 移动浏览器

移动浏览器	受支持的版本
iOS 版 Chrome、iOS 版 Safari	两个主要版本（当前版本和最新版本）  （不支持超低延迟播放。支持正常延迟播放。此限制条件适用于所有 iOS 浏览器。）  （仅播放器 1.3.0 和更高版本支持定时元数据。）
Chrome for iPadOS , Safari for iPadOS	两个主要版本（当前版本和最新版本）  （选择“请求移动网站”时：  • 不支持超低延迟播放。 • 仅播放器 1.3.0 和更高版本支持定时元数据。）
Android 版 Chrome	两个主要版本（当前版本和最新版本）

## 本机平台

平台	受支持的版本	支持的设备
Android	5.0 (Lollipop) 及更高版本	手机和平板电脑
iOS	12.0 和更高版本	全部

IVS 至少支持 4 个主要 iOS 版本和 6 个主要 Android 版本。我们当前版本的支持可能会超出这些最低限度。如果主要版本不再受支持，将至少提前 3 个月通过 SDK 发布说明通知客户。

## 减少第三方播放器的延迟

对于基本和标准通道类型：为了尽可能减少延迟，您必须使用 Amazon IVS 播放器。在第三方播放器（包括 iOS Safari）中，您可以使用以下配置将延迟减少到大约 10 秒：

- 将编码器（例如 OBS）的关键帧间隔设置为 2 秒或以下。
- 将 `?keyframeInterval=2` 添加到 RTMP/RTMPS URL。例如：`rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/sk_us-west-2_abcd1234efgh5678ijkl?keyframeInterval=2`

注意：作为 RTMP URL 一部分指定的关键帧间隔必须大于或等于在编码器中配置的值；否则，您可能会遇到播放问题。您可以将该值设置为 2 到 6 之间（含 2 和 6）的任意整数，但是 2 可以实现最低延迟。

对于高级通道类型：上述指南不适用。高级通道类型会自动生成关键帧间隔以提高编码效率，无论源编码关键帧间隔设置如何，关键帧之间的间隔最多为 2 秒。

## iOS Safari

在 iOS Safari 中，您可以使用 IVS 播放器并将其配置为使用服务工作线程，从而将延迟减少到大约 6-8 秒。请参阅 Player SDK：Web 指南中的[设置服务工作线程](#)，以了解实现细节和参考示例。

注意：要获得最低延迟，需要关键帧间隔设置为 2 秒的 IVS 流。



## 纯音频播放

所有 IVS 频道类型都支持纯音频播放。这对于移动应用程序尤其有价值。例如，在您的移动应用程序中，当用户对应用程序进行背景设置以节省带宽时，您可以将播放器切换到仅限音频的格式副本。

对于 ADVANCED-SD 和 ADVANCED-HD 频道，纯音频格式会自动包含在多变体播放列表中。对于 BASIC 和 STANDARD 频道，您必须将 `?allow_audio_only=true` 查询参数附加到播放 URL 才能包含纯音频格式副本。

注意：IVS 网络播放器 SDK 仅在 1.24.0 及更高版本中支持纯音频播放。

## 支持

如果您的视频流出现播放错误或其他播放问题，请通过播放器 API 确定唯一的播放会话标识符。

对于以下版本 Amazon IVS 播放器：	使用：
Android	<code>sessionId</code> 函数
iOS	IVSPlayer 的属性 <code>sessionId</code>
Web	<code>getSessionId</code> 函数

与 Amazon Support 共享此播放会话标识符。利用它可获得有助于解决问题的信息。

注意：播放器在不断改进。请参阅 [Amazon IVS 发布说明](#) 了解可用版本和已修复问题。如果合适，请在联系支持部门之前更新您的播放器版本，看看这是否解决了您的问题。

## 版本控制

Amazon IVS 播放器软件开发工具包使用 [语义化版本](#)。

在此讨论中，假设：

- 最新版本是 4.1.3。
- 先前主要版本的最新版本为 3.2.4。
- 版本 1.x 最新版本是 1.5.6。

最新版本的次要版本已添加向后兼容的新功能。在本例中，版本 4.2.0 已添加新功能。

最新版本的补丁版本已添加向后兼容、次要错误修复。在这里，版本 4.1.4 已添加次要错误修复。

向后兼容、主要错误修复处理方式不同；将在以下几个版本中添加：

- 最新版本补丁版本。在本例中是版本 4.1.4。
- 先前次要版本的补丁版本。在本例中是版本 3.2.5。
- 最新版本 1.x 版本的补丁版本。在本例中是版本 1.5.7。

主要错误修复由 Amazon IVS 产品团队定义。典型示例包括关键安全更新和客户所需的其他选定修复。

注意：在上面的例子中，发布的版本递增但不会跳过任何数字（例如，从 4.1.3 到 4.1.4）。实际上，一个或多个补丁编号可能保留在内部而不发布，因此发布版本可以从 4.1.3 增加到 4.1.6。

## Amazon IVS 播放器开发工具包：Web 指南

适用于 Web 的 Amazon Interactive Video Service (IVS) 播放器可以与[播放器框架](#)（如 Video.js）集成，或者在 HTML `<video>` 元素上独立使用。

最新版本的网络播放器：1.24.0（[发行说明](#)）

参考文档：有关亚马逊 IVS 网络播放器中可用的最重要方法的信息，请参阅参考文档 <https://aws.github.io/amazon-ivs-player-docs/1.24.0/web/>。

### 开始使用

我们通过 `script` 标签以及通过 npm 模块提供支持。

### 演示

以下现场演示展示了如何将 Web 播放器与来自我们的“内容分发网络：[Amazon IVS 播放器示例](#)”中的 `script` 标签一起使用。

### 使用脚本标签进行设置

要使用 `script` 标签设置 Amazon IVS 播放器，请执行以下操作：

1. 包含以下标签（适用于最新版本的播放器）。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-player.min.js">
```

2. 加载 `amazon-ivs-player.min.js` 后，它会为全局上下文添加一个 `IVSPlayer` 变量。这是您将用来创建播放器实例的库。首先，检查 `isPlayerSupported` 以确定浏览器是否支持 IVS 播放器：

```
if (IVSPlayer.isPlayerSupported) { ... }
```

然后，要创建播放器实例，请调用 `IVSPlayer` 对象上的 `create` 函数。

```
const player = IVSPlayer.create();
```

适用于 Web 的 Amazon IVS Player 开发工具包使用 Web 工作线程来优化视频播放。

3. 使用 `load` 和 `play` 函数在播放器实例上加载和播放流：

```
player.load("PLAYBACK_URL");  
player.play();
```

其中 `PLAYBACK_URL` 是请求流密钥时从 Amazon IVS API 返回的 URL。

## 示例代码

在此示例中，将 `PLAYBACK_URL` 替换为您要加载的源流。此示例使用最新版本的 Amazon IVS 播放器。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-player.min.js"></script>  
<video id="video-player" playsinline></video>  
<script>  
  if (IVSPlayer.isPlayerSupported) {  
    const player = IVSPlayer.create();  
    player.attachHTMLVideoElement(document.getElementById('video-player'));  
    player.load("PLAYBACK_URL");  
    player.play();  
  }  
</script>
```

在 `<video>` 标签中，要在 iOS Safari 上进行内联播放，必须有 `playsinline`。见 <https://webkit.org/blog/6784/new-video-policies-for-ios/>。

## 使用 NPM 进行设置

有关指导，包括示例 Webpack 配置文件，请参阅以下存储库：<https://github.com/aws-samples/amazon-ivs-player-web-sample>。

注意：托管来自您自己网域的玩家静态资产时，必须将 WebAssembly 二进制 (amazon-ivs-wasmworker.min.wasm) 的“内容类型”响应标头设置为“application/wasm”。您还应该对您的资产进行 gzip 压缩，以减少通过线路下载的字节，并缩短播放器开始播放的时间。

## TypeScript

如果你正在使用 TypeScript，npm 包中包含你可能想要导入和使用的类型。有关这些类型的信息，请参阅 [Amazon IVS 播放器开发工具包：Web 参考](#)。

## 设置服务工作线程

为了进一步降低通过仅支持本机播放的浏览器（主要是 iOS Safari）播放时的延迟，可以设置和配置服务工作线程。要了解更多信息，请参阅[减少第三方播放器的延迟](#)。

设置 Amazon IVS 播放器以使用服务工作线程：

1. 创建一个文件以从 CDN 卸载 IVS 服务工作线程。这是必需操作，因为服务工作线程必须托管在与引入他们的页面相同的域中。

创建名为 `amazon-ivs-service-worker-loader.js` 或类似值的文件，并添加以下行：

```
importScripts('https://player.live-video.net/1.24.0/amazon-ivs-service-worker.min.js');
```

2. 创建播放器实例时，请传入以下引用 `amazon-ivs-service-worker-loader.js` 文件的以下 `serviceWorker` 配置：

```
const player = IVSPlayerPackage.create({
  serviceWorker: {
    url: 'amazon-ivs-service-worker-loader.js'
  }
});
```

3. 在视频元素上，将 `crossOrigin` 属性设置为 `anonymous`。这是允许服务工作线程对清单进行更改所必需的。

注意：要在本地测试服务工作线程，需要通过 localhost 或 https 提供该页面。

有关实时演示，请参阅以下存储库中的服务工作线程示例：

<https://github.com/aws-samples/amazon-ivs-player-web-sample>

## 纯音频播放

必须使用该 `setQuality()` 方法手动选择纯音频质量。请注意，玩家不支持第二个参数的 `true` 值 `adaptive`，因此默认情况下，此参数是 `false`。

要在开始播放之前将音质设置为纯音频，请在 `READY` 事件 `setQuality()` 内部调用：

```
player.addEventListener(PlayerState.READY, () => {
  const qualities = player.getQualities();
  const audioOnly = qualities.find(q => q.name === 'audio_only');
  if (audioOnly) {
    player.setQuality(audioOnly);
  }
});
```

在其中设置画质 `READY` 适用于自动播放和非自动播放模式。

## 框架集成

适用于 Web 的 Amazon IVS 播放器开发工具包旨在轻松与您选择的框架集成。我们提供官方的 Video.js 集成（Video.js 行话表示为“技术”）。

以下是我们提供的 Web 播放器的简要比较：

播放器类型	描述	UI	插件
适用于 Web 的 Amazon IVS 播放器开发工具包	一个轻量级和可定制的选项，适合需要更多控制的开发人员。	否	否
<a href="#">适用于 Video.js 的 Amazon IVS 播放器技术</a>	一个功能齐全的选项，如果您已经使用 Video.js 并且想要一个交钥匙解决方案，这非常适合。	是 ( <a href="#">Video.js 皮肤</a> )	是 ( <a href="#">Video.js 插件</a> )

播放器类型	描述	UI	插件
<a href="#">适用于 JW 播放器的 Amazon IVS 播放器提供程序</a>	一个功能齐全的选项，如果您已经使用 JW 播放器并想要一个交钥匙解决方案，这非常适合。	是	不适用

## 使用内容安全策略

Amazon IVS Web 播放器开发工具包配置为在使用内容安全策略 (CSP) 的页面上工作。必须制定几个关键的 CSP 指令。在这里，我们描述了必需的最小指令集。可能需要额外的指令和来源，具体取决于您的具体设置。

以下是 CSP 所需的最低指令：

```
worker-src blob;;
media-src blob;;
connect-src *.live-video.net;
script-src 'wasm-unsafe-eval';
```

注意：较早版本的浏览器可能无法识别上述 CSP 规则中的一项或多项（例如 `wasm-unsafe-eval`），反而可能需要非常宽松的 CSP 策略（`unsafe-eval`）。但是，这与 CSP 限制在页面上运行危险 JavaScript 的要点背道而驰。我们建议的解决方法是将库资源托管在与页面相同的源上。

## 已知问题和解决方法

- 在 iOS 移动浏览器（例如 Safari 或 Chrome）上播放录制的内容（也称为 VOD）时，向后搜索将使播放器静音。

解决办法：在搜索之后调用 `player.setMuted(false)`。

- 在 iOS 移动浏览器上播放录制的内容时，若直接选择所需位置，向后搜索会间歇性地工作。

解决办法：将搜索栏拖动到所需位置。

- 在 iOS 移动浏览器上播放录制内容时，`player.seekTo()` 调用无法持续工作。

解决办法：在 `loadeddata` 事件之后，在视频 HTML 元素上设置 `currentTime`。例如：

```
videoEl.addEventListener('loadeddata', () => {
  videoEl.currentTime = 30; // seek 30s from the beginning
});
```

```
});
```

- 在 iOS 移动浏览器上播放实时流或录制的内容时，字幕可能不会以不同的大小呈现，并且可能会多次重新渲染。

解决办法：尚无。

- 在 iOS 移动浏览器上播放实时流或录制的内容时，`player.getQualities()` 调用不会返回可用质量列表。

解决办法：尚无。播放器只支持 iOS 浏览器上的自动质量模式。

- 启用本机 HTML5 控件后，调用 `setQuality()` 将被忽略。

解决办法：在调用 `player.setQuality()` 之前禁用 HTML5 控件。

- 在 iOS 移动浏览器上播放静音实时流时，如果恢复处于非活动状态的播放器选项卡（例如，选项卡切换或设备锁定/解锁），可能会出现播放器不稳定的情况（例如，黑屏或冻屏、缓冲）。

解决办法：使用 JavaScript [页面可见性 API](#) 检测页面可见性变化，然后对玩家采取相应的措施。例如：

```
//if client platform is iOS
if (!!navigator.platform && /iPad|iPhone|iPod/.test(navigator.platform)) {
  document.addEventListener("visibilitychange", () => {
    if (document.visibilityState === "hidden" && player.isMuted()) {
      player.pause()
    }
    if (document.visibilityState === "visible" &&
        player.getState() !== PlayerState.PLAYING) {
      player.play()
    }
  })
}
```

## Amazon IVS 播放器开发工具包：Android 指南

Amazon Interactive Video Player (IVS) Android 播放器开发工具包提供了在 Android 上使用 Amazon IVS 播放器所需的接口。

我们仅保证 Android 移动设备（手机和平板电脑）的播放性能。我们不支持 Android TV、Fire TV、IoT 设备和仿真器。

`com.amazonaws.ivs.player` 软件包实现了本文档中所描述的接口。支持以下操作：

- 设置 ( 初始化 ) 播放器。
- 管理播放。
- 管理质量。
- 接收事件。
- 接收错误。

最新版本的安卓播放器：1.24.0 ( [发行说明](#) )

参考文档：有关亚马逊 IVS 安卓播放器中可用的最重要方法的信息，请参阅参考文档 <https://aws.github.io/amazon-ivs-player-docs/1.24.0/android/>。

示例代码：参见 Android 示例存储库，[网址为 GitHub：https://github.com/aws-samples/amazon-ivs-player-android-sample](https://github.com/aws-samples/amazon-ivs-player-android-sample)。

平台要求：开发环境需要 Android 5.0 (Lollipop) 或更高版本。

提供了一个适用于 Amazon IVS 播放器 SDK 的 React Native 包装器。有关代码和文档，请参阅 <https://github.com/aws/amazon-ivs-react-native-player>。

## 开始使用

### 安装库

要将 Amazon IVS Android 播放器库添加到您的 Android 开发环境中，请将该库添加到您模块的 build.gradle 文件 ( 如此处所示，适用于最新版本的 Amazon IVS 播放器 )。

```
repositories {
    mavenCentral()
}

dependencies {
    implementation 'com.amazonaws:ivs-player:1.24.0'
}
```

如要手动安装 SDK，也可从以下位置下载最新版本：

<https://search.maven.org/artifact/com.amazonaws/ivs-player>



## 创建播放器并设置事件侦听器

播放器接口是 `com.amazonaws.ivs.player.Player`。将其初始化，如下所示：

```
// Create a player instance
// <this> refers to the current Android Activity
player = Player.Factory.create(this);

// Set up to receive playback events and errors
player.addListener(this);
```

或者，通过使用 `PlayerView` 初始化：

```
// Create a player instance
// <this> refers to the current Android Activity
PlayerView playerView = new PlayerView(this);
Player player = playerView.getPlayer();
// Set up to receive playback events and errors
player.addListener(this);
```

注意：在 Android 应用程序的主线程中执行侦听器回调方法。

## 设置视频的 Surface View

如果未使用 `PlayerView`，请将 `SurfaceView` 添加到您的 Android UI 布局以显示视频。此 `Surface` 必须可用，然后才能播放任何视频流。您可以通过 `SurfaceHolder` 接口访问基础 `Surface`，该接口可通过调用 `getHolder()` 进行检索。（参见 [SurfaceView](#) 安卓开发者参考手册）。使用接收有关表面变化的事件（参见 [SurfaceHolder.Callback](#)）。`SurfaceHolder.Callback`

```
surfaceView = (SurfaceView) findViewById(R.id.surfaceView);
surfaceView.getHolder().addCallback(this);

@Override
public void surfaceCreated(SurfaceHolder holder) {
    this.surface = holder.getSurface();
    if (player != null) {
        player.setSurface(this.surface);
    }
}

@Override
```

```
public void surfaceDestroyed(SurfaceHolder holder) {
    this.surface = null;
    if (player != null) {
        player.setSurface(null);
    }
}
```

## 播放视频流

由于视频流是异步加载的，因此播放器必须处于 READY 状态，然后您的应用程序才可以调用 play 方法开始播放。使用 `Player.Listener` 接口来确定播放器何时处于就绪状态。

请看下面的示例代码：

```
player.load(Uri.parse(url));

@Override
public void onStateChanged(Player.State state) {
    switch (state) {
        case BUFFERING:
            // player is buffering
            break;
        case READY:
            player.play();
            break;
        case IDLE:
            break;
        case PLAYING:
            // playback started
            break;
    }
}
```

## 释放播放器

当不再使用播放器时，必须调用 `player.release()` 方法，以释放库使用的资源。通常，这是在播放器的 Activity 或 Fragment 的 `onDestroy` 回调中完成的。

```
@Override
protected void onDestroy() {
    super.onDestroy();
}
```

```
player.removeListener(this);
player.release();
}
```

调用 `player.release()` 方法后，将无法再使用播放器。

## 权限

Android 播放器开发工具包需要以下权限：

```
<uses-permission android:name="android.permission.INTERNET" />
```

此外，这些可选权限还可以改善播放体验：

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

## 线程安全

播放器 API 不是线程安全的。对播放器实例发起的所有调用都应来自同一个线程。

## 开发工具包大小

Amazon IVS 播放器开发工具包设计为尽可能轻便。有关开发工具包大小的最新信息，请参阅[发布说明](#)。

**重要提示：**在评估大小影响时，Android Studio 生成的 AAB/APK 的大小并不代表您下载到用户设备的应用程序的大小。Google Play 商店会执行优化以减小应用程序的大小。建议使用 [Android 应用程序包](#)为每个设备配置提供优化的应用程序。

## 已知问题和解决方法

- 安卓播放器 SDK 的运行时依赖于 4.x OkHttp 版本。由于 API 签名不匹配和向 OkHttp 后兼容性问题，使用 3.x OkHttp 版本可能会导致不稳定或崩溃。具体而言，该播放器依赖于 4.2.2 OkHttp 版本，但它应该与任何 4.x 版本兼容。

**解决办法：**使用应用程序的 4.x 版本 OkHttp 或将其 OkHttp 从应用程序中删除。

- 使用 Android 11 ( API 级别 30 ) 仿真器时，您可能会遇到视频布局问题 ( 特别是视频流的缩放 ) 。

解决办法：在真实设备上播放。

## Amazon IVS 播放器开发工具包：iOS 指南

Amazon Interactive Video Service (IVS) iOS 播放器提供了在 iOS 上使用 Amazon IVS 播放器所需的界面。

iOS 播放器的最新版本：1.24.0 ( [发行说明](#) )

参考文档：有关亚马逊 IVS iOS 播放器中可用的最重要方法的信息，请参阅参考文档 <https://aws.github.io/amazon-ivs-player-docs/1.24.0/ios/>。

示例代码：参见 iOS 示例存储库，[网址为 GitHub : https://github.com/aws-samples/amazon-ivs-player-ios-sample](https://github.com/aws-samples/amazon-ivs-player-ios-sample)。

平台要求：开发需要 Xcode 11 或更高版本。SDK 支持 iOS 12 及更高版本的部署目标，适用于物理设备和 iOS 模拟器。

提供了一个适用于 Amazon IVS 播放器 SDK 的 React Native 包装器。有关代码和文档，请参阅 <https://github.com/aws/amazon-ivs-react-native-player>。

### 开始使用

我们建议您通过集成播放器 SDK CocoaPods。(或者，您可以手动将框架添加至项目。)

**推荐：集成播放器 SDK (CocoaPods)**

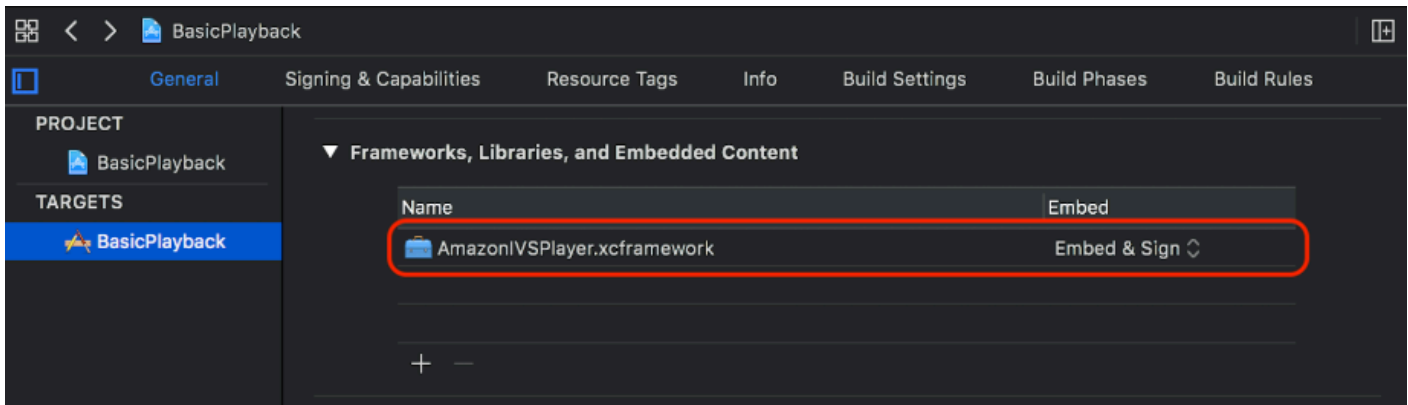
发行版以该 CocoaPods 名称发布 AmazonIVSPlayer。将此依赖项添加至您的 Podfile 中：

```
pod 'AmazonIVSPlayer'
```

运行 `pod install`，开发工具包将在 `.xcworkspace` 中可用。

**替代方法：手动安装框架**

1. 从 <https://player.live-video.net/1.24.0/AmazonIVSPlayer.xcframework.zip> 下载最新版本。
2. 提取归档的内容。AmazonIVSPlayer.xcframework 包含适用于设备和模拟器的开发工具包。
3. 通过以下方法嵌入 AmazonIVSPlayer.xcframework：将其拖动到应用程序目标 General (常规) 选项卡上的 Frameworks, Libraries, and Embedded Content (框架、库和嵌入式内容) 部分：



## 创建播放器

播放器对象是 `IVSPlayer`。按如下方法进行初始化：

### Swift

```
import AmazonIVSPlayer

let player = IVSPlayer()
```

### Objective-C

```
#import <AmazonIVSPlayer/AmazonIVSPlayer.h>

IVSPlayer *player = [[IVSPlayer alloc] init];
```

## 设置委托

委托回调提供有关回放状态、事件和错误的信息。所有回调都在主队列上调用。

### Swift

```
// Self must conform to IVSPlayer.Delegate
player.delegate = self
```

### Objective-C

```
// Self must conform to IVSPlayer.Delegate
player.delegate = self
```

## 显示视频

播放器在自定义层 `IVSPlayerLayer` 中显示视频。软件开发工具包还提供了 `IVSPlayerView`，自定义层支持的 `UIView` 子类。选择更方便您的应用程序界面的一种。

在这两种情况下，使用 `player` 属性，均可显示播放器实例中的视频。

### Swift

```
// When using IVSPlayerView:  
playerView.player = player  
  
// When using IVSPlayerLayer:  
playerLayer.player = player
```

### Objective-C

```
// When using IVSPlayerView:  
playerView.player = player;  
  
// When using IVSPlayerLayer:  
playerLayer.player = player;
```

## 加载流

播放器以异步方式加载流。其状态指示何时可以播放。

### Swift

```
player.load(url)
```

### Objective-C

```
[player load:url];
```

## 播放视频流

播放器准备就绪后，使用 `play` 按钮开始播放。使用 `state` 属性中的委托界面或键值观察来观察状态变化。以下是基于委托方法的示例：

## Swift

```
func player(_ player: IVSPlayer, didChangeState state: IVSPlayer.State) {
    if state == .ready {
        player.play()
    }
}
```

## Objective-C

```
- (void)player:(IVSPlayer *)player didChangeState:(IVSPlayerState)state {
    if (state == IVSPlayerStateReady) {
        [player play];
    }
}
```

## 暂停后台应用程序

播放器不支持后台播放，但不需要完全退出。暂停即可；请参阅下面的示例。

## Swift

```
override func viewDidLoad() {
    super.viewDidLoad()

    NotificationCenter.default.addObserver(self,
        selector: #selector(applicationDidEnterBackground(_:)),
        name: UIApplication.didEnterBackgroundNotification,
        object: nil)
}

@objc func applicationDidEnterBackground(_ notification: NSNotification) {
    playerView?.player?.pause()
}
```

## Objective-C

```
- (void)viewDidLoad {
    [super viewDidLoad];

    NotificationCenter *defaultCenter = NotificationCenter.defaultCenter;
```

```
[defaultCenter addObserver:self
                    selector:@selector(applicationDidEnterBackground:)
                    name:UIApplicationDidEnterBackgroundNotification
                    object:nil];
}

- (void)applicationDidEnterBackground:(NSNotification *)notification {
    [playerView.player pause];
}
```

## 线程安全

播放器 API 不是线程安全的。您应该从应用程序主线程创建和使用播放器实例。

## 组合起来

以下简单的视图控制器代码段在播放器视图中加载并播放 URL。请注意，playerView 属性是从 XIB/Storyboard 初始化而来，并且使用 [Identity Inspector](#) 中的“自定义类”部分在 Interface Builder 中将其类设置为 IVSPlayerView。

## Swift

```
import AmazonIVSPlayer

class MyViewController: UIViewController {
    ...
    // Connected in Interface Builder
    @IBOutlet var playerView: IVSPlayerView!

    override func viewDidLoad() {
        super.viewDidLoad()

        NotificationCenter.default.addObserver(self,
            selector: #selector(applicationDidEnterBackground(_:)),
            name: UIApplication.didEnterBackgroundNotification,
            object: nil)
    }

    @objc func applicationDidEnterBackground(_ notification: NSNotification) {
        playerView?.player?.pause()
    }
    ...
}
```



```

// Assumes this view controller is already loaded.
// For example, this could be called by a button tap.
func playVideo(url videoURL: URL) {
    let player = IVSPlayer()
    player.delegate = self
    playerView.player = player
    player.load(videoURL)
}
}

extension MyViewController: IVSPlayer.Delegate {
    func player(_ player: IVSPlayer, didChangeState state: IVSPlayer.State) {
        if state == .ready {
            player.play()
        }
    }
}
}

```

## Objective-C

```

// MyViewController.h

@class IVSPlayerView;

@interface MyViewController: UIViewController
...
// Connected in Interface Builder
@property (nonatomic) IBOutlet IVSPlayerView *playerView;
...
@end

// MyViewController.m

#import <AmazonIVSPlayer/AmazonIVSPlayer.h>

@implementation MyViewController <IVSPlayerDelegate>
...

- (void)viewDidLoad {
    [super viewDidLoad];

    NotificationCenter *defaultCenter = NotificationCenter.defaultCenter;

```

```
[defaultCenter addObserver:self
                    selector:@selector(applicationDidEnterBackground:)
                    name:UIApplicationDidEnterBackgroundNotification
                    object:nil];
}

- (void)applicationDidEnterBackground:(NSNotification *)notification {
    [playerView.player pause];
}

// Assumes this view controller is already loaded.
// For example, this could be called by a button tap.
- (void)playVideoWithURL:(NSURL *)videoURL {
    IVSPlayer *player = [[IVSPlayer alloc] init];
    player.delegate = self;
    playerView.player = player;
    [player load:videoURL];
}

- (void)player:(IVSPlayer *)player didChangeState:(IVSPlayerState)state {
    if (state == IVSPlayerStateReady) {
        [player play];
    }
}

...
@end
```

## 开发工具包大小

Amazon IVS 播放器开发工具包设计为尽可能轻便。有关开发工具包大小的最新信息，请参阅[发布说明](#)。

**重要提示：**在评估大小影响时，Xcode 生成的 IPA 的大小不代表下载到用户设备的应用程序的大小。应用商店执行优化以减小应用程序的大小。

## 已知问题和解决方法

- 在针对 arm64e 架构进行测试时，播放器可能会崩溃。这仅适用于专门针对 arm64e 进行测试的情况，不适用于应用商店构建。

**解决办法：**不要使用 arm64e。

# Amazon IVS 播放器开发工具包：Video.js 集成

本文档介绍了 Amazon Interactive Video Service (IVS) Video.js 播放器中最重要的功能。

Video.js 播放器集成的最新版本：1.24.0 ( [发行说明](#) )

## 开始使用

Amazon IVS 对 Video.js 的支持是通过 Video.js [技术](#)实现的。我们通过脚本标签以及 npm 模块提供支持。Amazon IVS 支持 Video.js 版本 7.6.6 以及更高的 7\* 和 8\* 版本。

请注意，在实例化播放器时，Video.js [源选项](#)不受支持。相反，请正常实例化播放器并调用 Video.js `src()` 函数。如果启用了自动播放，则流将开始播放；否则，使用 `play()` 以开始播放。

## 演示

以下现场演示展示了如何使用 Video.js 集成与我们的内容分发网络中的脚本标签：[Amazon IVS 播放器 Video.js 集成](#)。

## 使用脚本标签进行设置

要使用 `script` 标签设置 Amazon IVS 技术，请执行以下操作：

1. 包含以下标签（针对最新版本的播放器集成）。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-videojs-tech.min.js"></script>
```

2. 使用 `registerIVSTech` 函数注册技术：

```
registerIVSTech(videojs);
```

其中 `videojs` 是 Video.js 提供的对象。

3. 创建播放器的实例时，添加 AmazonIVS 作为 `techOrder` 选项中的第一个技术。

当实例化播放器时，Video.js [源选项](#)不受支持。相反，要设置源代码，请正常实例化播放器，然后调用 Video.js `src()` 函数。如果启用了自动播放，则流将开始播放；否则，使用 `play()` 以开始播放。

## 示例代码

在此示例中，`PLAYBACK_URL` 是要加载的源流。此示例使用最新版本的 Amazon IVS 播放器。

```
<!doctype html>
<html lang="en">
<head>
  <link href="https://cdnjs.cloudflare.com/ajax/libs/video.js/7.14.3/video-js.css"
  rel="stylesheet">
  <script src="https://cdnjs.cloudflare.com/ajax/libs/video.js/7.14.3/
video.min.js"></script>
  <script src="https://player.live-video.net/1.24.0/amazon-ivs-videojs-
tech.min.js"></script>
</head>

<body>
  <div class="video-container">
    <video id="amazon-ivs-videojs" class="video-js vjs-4-3 vjs-big-play-centered"
controls autoplay playsinline></video>
  </div>
  <style>
    body {
      margin: 0;
    }

    .video-container {
      width: 640px;
      height: 480px;
      margin: 15px;
    }
  </style>
  <script>
    (function play() {
      // Get playback URL from Amazon IVS API
      var PLAYBACK_URL = '';

      // Register Amazon IVS as playback technology for Video.js
      registerIVSTech(videojs);

      // Initialize player
      var player = videojs('amazon-ivs-videojs', {
        techOrder: ["AmazonIVS"]
      }, () => {
        console.log('Player is ready to use!');
        // Play stream
        player.src(PLAYBACK_URL);
      });
    });
```

```
    })();  
  </script>  
</body>  
</html>
```

## 使用 NPM 进行设置

要通过 npm 使用 Amazon IVS 播放器，请执行以下操作：

1. 安装 [video.js](#) npm 软件包或确保您的项目具有对 Video.js 库的一些其他访问权限。
2. 安装 `amazon-ivs-player` npm 软件包：

```
npm install amazon-ivs-player
```

3. 准备好注册 Amazon IVS 技术时，请导入 `registerIVSTech` 函数：

```
import { registerIVSTech } from 'amazon-ivs-player';
```

4. 使用 `registerIVSTech` 函数注册技术：

```
registerIVSTech(videojs, options);
```

其中：

- `videojs` 是 Video.js 提供的对象。
- `options` 是 Amazon IVS 技术层的选项。支持的选项为：
  - `wasmWorker`：托管 `amazon-ivs-wasmworker.min.js` 文件的 URL。
  - `wasmBinary`：托管 `amazon-ivs-wasmworker.min.wasm` 文件的 URL。

工作线程文件位于 `amazon-ivs-player/dist/` 下的 `node_modules/` 文件夹中。您需要托管它们以使用 IVS 播放器。

5. 创建播放器的实例时，添加 AmazonIVS 作为 `techOrder` 选项中的第一个技术：

```
const player = videojs('videojs-player', {  
  techOrder: ["AmazonIVS"]  
});
```

## TypeScript

如果你正在使用 TypeScript，我们的 npm 包包括你可能想要导入和使用的以下类型。

- VideoJSEvents，描述了从 `getIVSEvents()` 返回的结构。
- VideoJSIVSTech，描述了使用 Amazon IVS 技术的播放器实例的接口。此值可以与 [@types/video.js](#) npm 软件包公开的 VideoJsPlayer 类型[相交](#)。
- TechOptions，描述了定义您可以发送到 `registerIVSTech()` 的配置选项的接口。

有关这些类型的更多信息，请参阅 [Amazon IVS 播放器开发工具包：Web 参考](#)。

## 事件

要侦听标准 Video.js 事件，请使用 Video.js 播放器的 [on](#) 函数。

要侦听特定于 Amazon IVS 的事件，请在 Amazon IVS Web 播放器上添加和删除事件侦听器：

```
player.getIVSPlayer().addEventListener(event, callback);
player.getIVSPlayer().removeEventListener(event, callback);
```

其中 `callback` 是您定义的回调，`event` 为以下值之一：`PlayerEventType` 或 `PlayerState`。有关事件的更多信息，请参阅 [Amazon IVS 播放器开发工具包：Web 参考](#)。

## 错误

对于一般 Video.js 错误，请在播放器上侦听通用 `error` 事件：

```
player.on("error", callback);
```

对于特定于 Amazon IVS 的错误，请在 Amazon IVS 播放器上侦听其自身的错误：

```
let playerEvent = player.getIVSEvents().PlayerEventType;
player.getIVSPlayer().addEventListener(playerEvent.ERROR, callback);
```

回调将接收带有以下字段的对象：

字段	描述
type	错误类型。对应于 <code>ErrorType</code> 事件。有关更多信息，请参阅 <a href="#">Amazon IVS 播放器开发工具包：Web 参考</a> 。
code	错误代码。
source	错误源。
message	人类可读的错误消息。

## 插件

我们提供了一个插件，可为可用质量创建 UI 切换。要使用此插件，如果您通过以下 `script` 标签使用我们的技术，则必须通过包含 `amazon-ivs-quality-plugin.min.js` 文件来加载该插件（对于最新版本的 IVS 播放器）：

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-quality-plugin.min.js"></script>
```

如果您使用的是 npm，请导入来自 `amazon-ivs-player` 模块的 `registerIVSQualityPlugin`：

```
import { registerIVSQualityPlugin } from 'amazon-ivs-player';
```

创建 `Video.js` 播放器的实例后，需要执行以下调用才能进行注册并启用：

```
registerIVSQualityPlugin(videojs); // where videojs is the video.js variable  
player.enableIVSQualityPlugin(); // where player is the instance of the videojs player
```

这将创建一个 UI 菜单按钮，允许您选择流的质量。

## 插件和 TypeScript

如果你正在使用 TypeScript，我们的 npm 包中包含你可能想要导入的 `VideoJSQualityPlugin` 类型，并与我们的插件一起使用。插件本质上是 `mixin`，因此该类型接口被用作与 `VideoJSIVSTech` TypeScript 接口的 [相交类型](#)。

## 内容安全策略

Amazon IVS Video.js API 配置为在使用内容安全策略 (CSP) 的页面上工作。请参阅 [Amazon IVS 播放器开发工具包：Web 指南](#) 中的“使用内容安全策略”章节。

## 函数

### 播放

Amazon IVS Video.js API 支持 Video.js 框架内部使用的必要接口。客户端应用程序不太可能需要直接使用这些方法，因为 Video.js 进行了必要的集成并提供了一个标准接口。但是，如果需要，访问内部 Video.js 和 Amazon IVS 播放器方法的一种方式是使用 Video.js 播放器对象获取技术所需的对象句柄。

要访问 API，请像平常一样检索 Video.js 播放器的实例：

```
let player = videojs("videoTagId"); //replace videoTagId with your <video> tag's id
```

然后，可以在该实例上调用函数。

以下是 Amazon IVS 技术层覆盖的 Video.js 函数的子集。有关 Video.js 函数的完整列表，请参阅 [video.js API 文档](#)。

函数	描述和 Amazon IVS 特定信息
<a href="#">currentTime</a>	获取或设置时间（以秒为单位，从开始算起）。  Amazon IVS：我们不建议为实时流设置当前时间。
<a href="#">dispose</a>	删除播放器实例  Amazon IVS：这也会删除 Amazon IVS 技术后端。
<a href="#">duration</a>	返回视频的持续时长（以秒为单位）。  Amazon IVS：对于实时流，这将返回 Infinity。
<a href="#">load</a>	开始加载 src() 数据。  Amazon IVS：无操作。



函数	描述和 Amazon IVS 特定信息
<a href="#">play</a>	<p>播放通过 <code>src</code> 调用设置的流。</p> <p>Amazon IVS：如果实时流已暂停，则播放最新帧的实时流，而不是从暂停的位置继续播放。</p>
<a href="#">playbackRate</a>	<p>获取或设置视频播放速率。1.0 表示正常速度；0.5 表示正常速度的一半；2.0 表示正常速度的两倍；依此类推。</p> <p>Amazon IVS：在实时流上，<code>get</code> 返回 1，并忽略一个集合。</p>
<a href="#">seekable</a>	<p>返回可以搜索到的媒体的 <code>TimeRanges</code>。</p> <p>Amazon IVS：对于实时流，对返回值 (<code>TimeRange</code>) 调用 <code>end(0)</code> 将返回无穷大。</p>

## Amazon IVS 特定

Amazon IVS Video.js 技术具有用于访问特定于 Amazon IVS 功能的行为的附加函数：

函数	描述
<a href="#">getIVSPlayer</a>	<p>返回底层 Amazon IVS 播放器实例。完整的 Amazon IVS 播放器 Web API 可通过此实例获得。我们建议尽可能使用基本的 Video.js 播放 API，并仅使用此函数访问 Amazon IVS 特定的功能。您可能需要在 Amazon IVS 播放器实例上访问的最常见函数是 <code>setQuality()</code> 和 <code>addEventListener()</code> / <code>removeEventListener()</code>。</p>
<a href="#">getIVSEvents</a>	<p>返回一个包含 Amazon IVS 特定枚举的对象。这用于侦听 Amazon IVS 特定错误。有关更多信息，请参阅<a href="#">事件</a>和<a href="#">错误</a>。</p>

## currentTime

获取或设置时间（以秒为单位，从开始算起）。

Amazon IVS：我们不建议为实时流设置当前时间。

## Signatures

```
currentTime  
currentTime(time)
```

### 参数

参数	类型	描述
time	number	如果 time 不存在，则获取当前时间。如果 time 存在，则将视频播放设置为该时间。

### 返回值

类型	描述
number	当前时间（以秒为单位，从开始算起）。

## dispose

删除播放器实例。

Amazon IVS：这也会删除 Amazon IVS 技术后端。

### 签名

```
dispose()
```

### 参数

无

### 返回值

无

## duration

返回视频的持续时长（以秒为单位）。

Amazon IVS：对于实时流，这将返回 Infinity。

### 签名

```
duration()
```

### 参数

无

### 返回值

类型	描述
number	流的持续时间，以秒为单位。对于实时流，此值为 Infinity。

## getIVSEvents

返回一个包含 Amazon IVS 特定枚举的对象。这用于侦听 Amazon IVS 特定的错误和事件。有关更多信息，请参阅：

- 此文档中的 [事件](#) 和 [错误](#)。
- 有关事件、错误类型和错误源的详细信息，请参阅 [Amazon IVS 播放器开发工具包：Web 参考](#)。

### 签名

```
getIVSEvents()
```

### 参数

无

## 返回值

类型	描述
object	具有 <code>PlayerEventType</code> 、 <code>PlayerState</code> 和 <code>ErrorType</code> 键的对象，这些键映射到其关联的枚举。

## getIVSPlayer

返回底层 Amazon IVS 播放器实例。完整的 Amazon IVS 播放器 Web API 可通过此实例获得。我们建议尽可能使用基本的 Video.js 播放 API，并仅使用此函数访问 Amazon IVS 特定的功能。您可能需要在 Amazon IVS 播放器实例上访问的最常见函数是 `setQuality()` 和 `addEventListener()` / `removeEventListener()`。

### 签名

```
getIVSPlayer()
```

### 参数

无

## 返回值

类型	描述
<code>MediaPlayer</code>	已创建的播放器实例。

## load

开始加载 `src()` 数据。

Amazon IVS：无操作。

## 签名

```
load()
```

## 参数

无

## 返回值

无

## play

播放通过 src 调用设置的流。

Amazon IVS：如果实时流已暂停，则播放最新帧的实时流，而不是从暂停的位置继续播放。

## 签名

```
play()
```

## 参数

无

## 返回值

无

## playbackRate

获取或设置视频播放速率。1.0 表示正常速度；0.5 表示正常速度的一半；2.0 表示正常速度的两倍；依此类推。

Amazon IVS：在实时流上，get 返回 1，并忽略一个集合。

## Signatures

```
playbackRate  
playbackRate(rate)
```

## 参数

参数	类型	描述
rate	number	播放速率。有效值：在 [0.25, 2.0] 范围内。

## 返回值

类型	描述
number	播放速率。

## seekable

返回可以搜索到的媒体的 TimeRanges。

Amazon IVS：对于实时流，对返回值 (TimeRange) 调用 `end(0)` 将返回无穷大。

## 签名

```
seekable()
```

## 参数

无

## 返回值

类型	描述
TimeRange	TimeRange 有哪些媒体可供寻找。

## Amazon IVS 播放器开发工具包：JW 播放器集成

本文档介绍了 Amazon Interactive Video Service (IVS) JW 播放器集成中最重要的功能。

JW Player 集成的最新版本：[1.24.0 \(发行说明\)](#)

## 开始使用

Amazon IVS 对 JW 播放器的支持是通过提供商实施的。Amazon IVS 提供程序仅在 JW 播放器的 Web 播放器上受支持。提供程序是通过脚本标签加载的，任何需要 Amazon IVS 提供程序播放的流都必须使用播放列表中的 `type: 'ivs'` 进行标记。Amazon IVS 支持 JW 播放器版本 8.18.4 及更高版本。

## 设置

在这些说明中, `JW_PLAYER_DIV` 为 JW 播放器实例 `<div>` 的名称, `IVS_STREAM` 是您的 IVS 播放 URL。要设置 Amazon IVS 提供程序并启用播放, 请执行以下操作：

1. 包括以下 `script` 标签 (适用于最新版本的播放器集成；在本例中为 1.24.0)：

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-jw-provider.min.js"></script>
```

2. 使用 `ivs` 类型来标记 IVS 播放列表项。将 `setup()` 中的 `cast` 值设置为 `null` (因为不支持 Chromecast)。

```
jwplayer(JW_PLAYER_DIV).setup({
  playlist: [{
    file: IVS_STREAM,
    type: 'ivs',
  }]
});
```

3. 如果您希望引用底层 Amazon IVS 播放器进行 Amazon IVS 播放器 API 调用, 或者您想要引用 Amazon IVS 特定枚举以进行回调处理, 请将侦听器添加到 `'providerPlayer'` 事件：

```
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
  // player object has 'ivsPlayer' and 'ivsEvents' properties
  // ...callback code...
});
```

## 示例代码

在此示例中, `JW_PLAYER_LIB` 是指向您的 JW 播放器库脚本的 URL, `IVS_STREAM` 是您的 IVS 播放 URL。

```
<!DOCTYPE html>
<html lang="en">
<head>
  <script src=JW_PLAYER_LIB></script>
  <script src="https://player.live-video.net/1.24.0/amazon-ivs-jw-provider.min.js"></
script>
</head>
<body>
  <div id='player'></div>
  <script>
    // set default values for ivsPlayer and ivsEvents
    var ivsPlayer = {};
    var ivsEvents = {};

    // define our player setup
    const ivsConfig = {
      playlist: [{
        file: IVS_STREAM,
        type: 'ivs',
      }]
    };

    jwplayer('player')
      .setup(ivsConfig)
      .on('providerPlayer', function (player) {
        console.log('Amazon IVS Player: ', player.ivsPlayer);
        console.log('Amazon IVS Player Events: ', player.ivsEvents);

        // store the reference to the Amazon IVS Player
        ivsPlayer = player.ivsPlayer;
        // store the reference to the Amazon IVS Player Events
        ivsEvents = player.ivsEvents;
      });
  </script>
</body>
</html>
```

## 事件

要收听标准 JW 播放器事件，请使用 JW 播放器的[开启](#)功能。

要收听特定于 Amazon IVS 的事件，或者添加和删除 Amazon IVS Web 播放器上的事件侦听器，您必须收听 'providerPlayer' 事件来获取对 Amazon IVS 播放器的引用，然后添加侦听事件。例如：



```
// store a default value for ivsPlayer
var ivsPlayer = {};

// store references to the Amazon IVS Player and Amazon IVS Events:
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
    ivsPlayer = player.ivsPlayer;
});

// set up event listening
ivsPlayer.addEventListener(event, callback);
ivsPlayer.removeEventListener(event, callback);
```

其中 `callback` 是您定义的回调，`event` 为以下值之一：`PlayerEventType`、`PlayerState` 或 `ErrorType`。有关事件的更多信息，请参阅 [Amazon IVS 播放器开发工具包：Web 参考](#)。

'providerPlayer' 事件由 JW 播放器发出，并且您注册的回调将收到一个包含以下字段的对象：

字段	描述
<code>ivsPlayer</code>	返回底层 Amazon IVS 播放器实例。完整的 Amazon IVS 播放器 Web API 可通过此实例获得。我们建议尽可能使用基本的 JW 播放器播放 API，并且仅使用此功能访问特定于 Amazon IVS 的功能。您可能需要在 Amazon IVS 播放器实例上访问的最常见函数是 <code>addEventListener()</code> 和 <code>removeEventListener()</code> 。
<code>ivsEvents</code>	返回带有 <code>PlayerEventType</code> 、 <code>PlayerState</code> 和 <code>ErrorType</code> 字段的对象，这些字段映射到其关联的 Amazon IVS 特定枚举。有关更多信息，请参阅 <a href="#">Amazon IVS 播放器开发工具包：Web 参考</a> 。

## 错误

对于常规 JW 播放器错误，请使用 JW 播放器的 [启用](#) 功能来侦听错误事件。

对于特定于 Amazon IVS 的错误，请在 Amazon IVS 播放器上侦听其自身的错误：

```
// set default values for ivsPlayer and ivsEvents
var ivsPlayer = {};
var ivsEvents = {};
```

```
// store references to the Amazon IVS Player and Amazon IVS Events
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
  ivsPlayer = player.ivsPlayer;
  ivsEvents = player.ivsEvents;
});

// set up event listening:
let playerEvent = ivsEvents.PlayerEventType;
ivsPlayer.addEventListener(playerEvent.ERROR, callback);
```

回调将接收带有以下字段的对象：

字段	描述
type	错误类型。对应于 <code>ErrorType</code> 事件。有关更多信息，请参阅 <a href="#">Amazon IVS 播放器开发工具包：Web 参考</a> 。
code	错误代码。
source	错误源。
message	人类可读的错误消息。

## 内容安全策略

Amazon IVS 提供程序 API 配置为在使用内容安全策略 (CSP) 的页面上工作。请参阅 [Amazon IVS 播放器开发工具包：Web 指南](#) 中的“使用内容安全策略”章节。

## 限制

此提供程序不支持转换。如果您在 JW 播放器控制面板中启用了转换功能，则可以在调用 `setup()` 时通过将 `cast` 设置为 `null` 禁用该功能。这隐藏了转换按钮。

## 将元数据嵌入视频流中

Amazon Interactive Video Service (IVS) 定时元数据提供了一种在 Amazon IVS 视频流中嵌入元数据的方法。可确保所有观众在视频流中同时接收元数据，而不考虑视频流延迟或地理位置。

### 什么是定时元数据？

定时元数据是带有时间戳的元数据。可使用 IVS API 或 IVS 广播 SDK 以编程方式将其插入流中。Amazon IVS 处理流时，定时元数据与音频和视频帧同步。在播放过程中，视频流的所有观众可同时获取视频流相关的元数据。时间码作为提示点，可用于根据数据触发操作，如下所示：

- 更新体育赛事的球员统计信息。
- 发送实时购物流的产品详细信息。
- 发送实时测验流的问题。

Amazon IVS 定时元数据使用嵌入在视频段中的 ID3 标签。因此，可以在录制的视频中使用。

### 设置 IAM 权限

先决条件：继续操作之前，您应已完成 [???](#) (包括创建 IAM 用户和设置权限)。

接下来，您必须授予 IAM 用户使用定时元数据的权限。按照以下步骤进行操作：

1. 登录 Amazon 管理控制台，并通过以下网址打开 IAM 控制台：<https://console.aws.amazon.com/iam/>。
2. 在导航窗格中，选择 Users (用户)，然后选择所需的用户 (在创建亚马逊云科技账户时指定的用户名)。
3. 在用户 Summary (摘要) 窗口中的 Permissions (权限) 选项卡上，选择 Add inline policy (添加内联策略) (右侧)。
4. 在 JSON 选项卡上，粘贴以下 Blob：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
    "Effect": "Allow",
    "Action": [
        "ivs:PutMetadata"
    ],
    "Resource": "arn:aws:ivs:*:*:channel/*"
}
]
```

5. 仍在 Create Policy (创建策略) 窗口中，选择 Review Policy (查看策略)。为策略指定一个 Name (名称)，然后选择 Create Policy (创建策略)。
6. 您将返回用户 Summary (摘要) 窗口，其中显示新的策略名称。

## 插入定时元数据

只能将定时元数据插入到指定通道上的活动流中。

### 使用 Amazon CLI

对于测试，添加定时元数据的最简单方法是使用 Amazon CLI。使用 Amazon CLI，需要首先在计算机上下载并配置 CLI。按照 [IVS 入门](#) 逐步进行操作时，您可能已完成如上操作；如果没有，请立即执行。有关详细信息，请参阅 [Amazon 命令行界面用户指南](#)。

有了 CLI 后：

1. 运行 put-metadata 命令并传入通道 ARN 和元数据：

```
aws ivs put-metadata --channel-arn <your-channel-arn> --metadata <your-metadata>
```

例如：

```
aws ivs put-metadata --channel-arn arn:aws:ivs:us-west-2:465369119046:channel/
GbiYJna5hFoC --metadata '{"question": "What does IVS stand for?", "correctIndex":
0, "answers": ["interactive video service", "interesting video service", "ingenious
video service"]}'
```

2. Amazon IVS 检查视频流是否为实时。如果视频流不是实时的，则会出现错误；否则，CLI 返回时不会出现错误，并将元数据（文本 blob）插入到视频流中。这种情况会尽快发生。无法保证何时发生；但所有观众都会在流视频中的同一点看到元数据。

## 使用 Amazon IVS API

要以编程方式插入定时元数据，请使用 [PutMetadata](#) API 端点。

以下是 HTTP 请求示例：

```
POST /PutMetadata HTTP/1.1
{
  "channelArn": "my_channel",
  "metadata": "{\"question\": \"What does IVS stand for?\", \"correctIndex\": 0, \"answers\": [\"interactive video service\", \"interesting video service\", \"ingenious video service\"]}"
}
```

## 使用 IVS 广播 SDK

您可以使用 IVS 广播 SDK 在带内插入定时元数据。这可能有助于将元数据与音频和视频内容同步。

- Android – 在 `BroadcastSession` 类中，请使用 `sendTimedMetadata`。
- iOS – 在 `IVSBroadcastSession` 类中，请使用 `sendTimedMetadata`。

## 使用定时元数据

借助 Amazon IVS 播放器，使用嵌入在视频流中的定时元数据。请参阅 [Amazon IVS 播放器开发工具包](#) 和文档的其余部分。

以下是使用 Amazon IVS 播放器软件开发工具包将接收到的任何元数据打印到控制台的示例代码段。播放包含嵌入元数据的片段时，都会触发一个事件。（Web 的事件是 `TEXT_METADATA_CUE`，Android 是 `onCue()`，iOS 是 `player(_:didOutputCue:)`。）您可以使用此事件在客户端应用程序中启动功能，例如更新交互式小部件。直播和录制内容都会触发此事件。

适用于 Web 的 Amazon IVS 播放器开发工具包：

```
const player = IVSPlayer.create();
player.addEventListener(IVSPlayer.PlayerEventType.TEXT_METADATA_CUE,
  function (cue) {
    console.log('Timed metadata: ', cue.text);
  });
```

适用于 Android 的 Amazon IVS 播放器开发工具包：

```
@Override
public void onCue(@NonNull Cue cue) {
    if(cue instanceof TextMetadataCue) {
        Log.i("Timed Metadata: ", ((TextMetadataCue)cue).text);
    }
}
```

适用于 iOS 的 Amazon IVS 播放器开发工具包：

```
func player(_ player: IVSPlayer, didOutputCue cue: IVSCue) {
    if let textMetadataCue = cue as? IVSTextMetadataCue {
        print("Timed Metadata: \(textMetadataCue.text)")
    }
}
```

注意：在播放器 1.3.0 及以上版本中，iOS Safari 和 iOS Chrome 支持定时元数据。

## 示例演示：测验应用程序

GitHub 上提供了交互式测验应用程序的代码示例。我们通过定时元数据使用 JSON 填充测验用户界面来显示问题和答案。答案是可选的，并显示选择是否正确。

Amazon IVS 播放器软件开发工具包平台	示例代码库
Web	<a href="https://github.com/aws-samples/amazon-ivs-basic-web-sample">https://github.com/aws-samples/amazon-ivs-basic-web-sample</a> 在此存储库中，请参阅 <a href="#">测验演示</a> （和 <a href="#">实际演示</a> ）。
Android	<a href="https://github.com/aws-samples/amazon-ivs-player-android-sample">https://github.com/aws-samples/amazon-ivs-player-android-sample</a> 在此存储库中，请参阅 <a href="#">测验演示</a> 。
iOS	<a href="https://github.com/aws-samples/amazon-ivs-player-ios-sample">https://github.com/aws-samples/amazon-ivs-player-ios-sample</a> 在此存储库中，请参阅 <a href="#">测验演示</a> 。

## 查看定时元数据

如需要，您可以在控制台中查看嵌入在实时流中的定时元数据：

1. 打开 [Amazon IVS 控制台](#)。
2. 在左上角选择汉堡包图标以打开导航窗格，然后选择 Live channels (实时通道)。
3. 选择要查看其流的通道，以转到该通道的详细信息页面。

此实时流正在该页面的 Live stream (实时流) 部分播放。

4. 在窗口底部，选择 Timed Metadata (定时元数据)。

播放器播放时，收到每个时间元数据事件时，会显示其值和收到的时间。

## 了解更多信息

请参阅 [Using Amazon Interactive Video Service Timed Metadata](#) (使用 Amazon Interactive Video Service 定时元数据)，这是关于使用 Amazon IVS 定时元数据的两部分博客系列中的第一部分。

## 设置私有通道

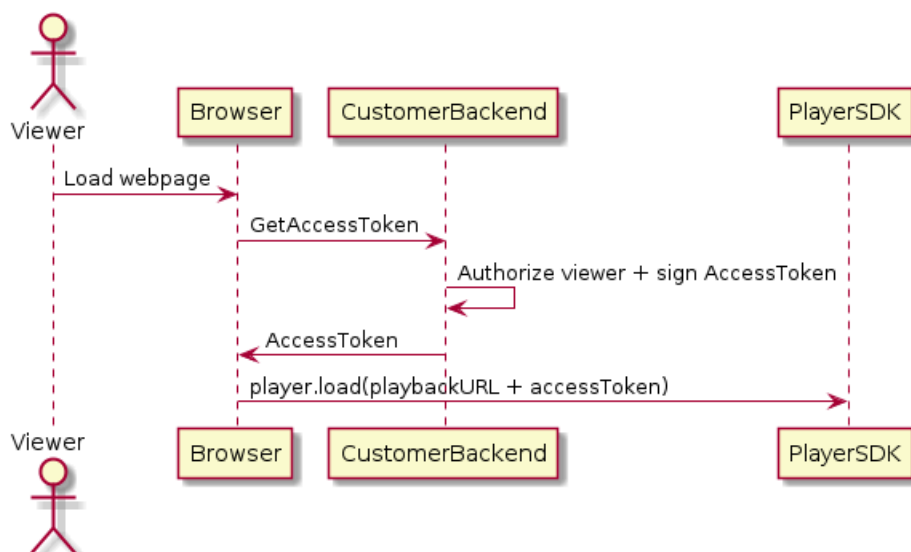
Amazon Interactive Video Service(IVS) 使客户能够创建私有通道，因此客户可以按通道或观众限制其视频流。客户可以通过以下方法控制对视频播放的访问：在通道上启用播放授权，并为授权的播放请求生成已签名的 JSON Web 令牌 (JWT)。

要求在通道上启用播放授权选项是可选项。观众尝试观看视频流时，如果通道启用了播放授权，Amazon IVS 会验证观众在请求中是否具有有效的播放令牌。播放令牌是一个 JWT，由 Amazon IVS 客户签名（使用播放授权密钥），并包含在启用了播放授权的通道的每个播放请求中。

### 主题

- [私有通道的工作流](#)
- [创建或导入播放密钥](#)
- [在通道上启用播放授权](#)
- [生成和签名播放令牌](#)
- [列出播放密钥](#)
- [删除播放密钥](#)
- [获取有关播放密钥的信息](#)
- [撤消查看者会话](#)

## 私有通道的工作流





1. 当观众尝试加载私有视频流的网页时，浏览器会请求访问令牌。（客户需要提供浏览器代码来执行此操作。）
2. 客户的后端应用程序接收访问令牌请求，并确定是否应授权该观众查看视频流。如果是，后端将生成 JWT，使用客户的私钥对其进行签名，并将播放请求中签名的 JWT 返回给浏览器。
3. 浏览器使用对 Amazon IVS 播放器（或其他播放器）开发工具包的请求来加载视频流。请求包含视频流播放 URL 和签名的 JWT。
4. Amazon IVS 使用客户的公钥来验证 JWT 是否使用了正确的私钥签名。
5. 如果 JWT 已通过验证，Amazon IVS 将为观众播放私有视频流。

客户负责创建：

- 请求访问令牌的浏览器代码。
- 生成和签名 JWT 的后端服务器应用程序。
- 播放授权密钥对。密钥对包括两部分：亚马逊云科技保留的公有密钥和您下载的私有密钥。使用私钥，您可以对授权访问您私有通道的 JWT 进行签名。

上述方法（使用浏览器的网络请求获取令牌）并不是实现播放授权的唯一方法。或者，客户可以在初始网页中发送已签名的播放令牌，以减少观众所需的网络往返次数。

在下面的章节中，我们将介绍如何将通道设为私有（启用播放授权）、生成和签名播放令牌以及使用播放密钥对。

注意：在以下控制台说明中，如果未显示左侧导航菜单，则可通过选择左上方的汉堡包图标将其打开。

## 创建或导入播放密钥

Amazon IVS 最多允许三个用于签名和验证播放令牌的密钥对。Amazon IVS 不提供任何密钥轮换。

导入后，无法更新播放密钥。必须删除现有的播放密钥并导入新的密钥。

您需要生成一个 [ECDSA 公有/私有密钥对](#) 以对 JWT 进行签名，并将公钥作为播放密钥资源上传到 Amazon IVS。然后，Amazon IVS 可以在播放请求中验证签名。

## 创建新的密钥对

创建密钥对的方法多种多样；下面我们举两个例子。

要在控制台中创建新的密钥对，请按照以下步骤操作。请注意，此过程仅允许您下载私有密钥。

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在左侧导航菜单中，选择“播放安全” > “播放键”。
3. 选择 Create playback key (创建播放密钥)。系统会显示 Create playback key (创建播放密钥) 对话框。
4. 按照提示操作并选择 Create (创建)。
5. Amazon IVS 将生成新的密钥对。公钥将作为回放密钥资源导入，私钥立即可供下载。

Amazon IVS 在客户端生成密钥，并且不存储私有密钥。请务必保存好密钥；您之后无法检索此密钥。

要使用 OpenSSL 创建新的 P384 EC 密钥对（您可能需要先安装 [OpenSSL](#)），请按照以下步骤操作。此过程允许您访问私有密钥和公有密钥。只有当您想测试令牌的验证时，才需要公有密钥。

```
openssl ecparam -name secp384r1 -genkey -noout -out priv.pem
openssl ec -in priv.pem -pubout -out public.pem
```

现在按照以下说明导入新的公有密钥。

## 导入现有公有密钥

如果您已经有密钥对，则可以将公有密钥导入 IVS。我们的系统不需要私有密钥，但您可以使用私有密钥来签署令牌。

要使用控制台导入现有公有密钥，请执行以下操作：

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在左侧导航菜单中，选择“播放安全” > “播放键”。
3. 选择 Import (导入)。系统会显示 Import playback key (导入播放密钥) 对话框。
4. 按照提示操作并选择 Import (导入)。
5. Amazon IVS 会导入您的公钥并生成播放密钥资源。

要使用 CLI 导入现有公有密钥，请执行以下操作：

```
aws ivs import-playback-key-pair --public-key-material "`cat public.pem`" --region
<aws-region>
```

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

以下是响应示例：

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:693991300569:playback-key/f99cde61-
c2b0-4df3-8941-ca7d38acca1a",
    "fingerprint": "98:0d:1a:a0:19:96:1e:ea:0a:0a:2c:9a:42:19:2b:e7",
    "tags": {}
  }
}
```

## API 请求

```
POST /ImportPlaybackKeyPair HTTP/1.1
{
  "publicKeyMaterial": "<pem file contents>"
}
```

## 在通道上启用播放授权

通道的授权要求可以在创建通道时或稍后（使用已更新的终端节点）进行配置。请注意，无论您要启用还是禁用播放授权，步骤都是相同的。

### 控制台说明

要在创建通道时启用授权，请执行以下操作：

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在 Get started (开始使用) 框（右上角）中，选择 Create channel (创建通道)。
3. 在 Create channel (创建通道) 页面上，选择 Custom configuration (自定义配置)。
4. 在 Playback authentication (播放身份验证) 部分，启用 Enable token-authentication requirement for video playback (启用视频播放令牌身份验证要求)。
5. 请按照接下来的提示来创建通道。（请参阅 [IVS 入门](#)。）

要通过更新现有通道启用授权，请执行以下操作：

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在左侧导航菜单中，选择 Channels (通道)。

3. 选中要更新的通道的复选框，然后选择 Edit (编辑)。
4. 在 Playback authentication (播放身份验证) 部分，启用 Enable token-authentication requirement for video playback (启用视频播放令牌身份验证要求)。
5. 单击 Save Changes (保存更改)。

## CLI 说明

要在创建通道时启用授权，请执行以下操作：

```
aws ivs create-channel --authorized --region <aws-region>
```

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

以下是响应示例。请注意，`authorized` 为 `true`。

```
{
  "streamKey": {
    "channelArn": "arn:aws:ivs:us-west-2:123456789:channel/fbc789c1-2c56-4ce6-a30a-d99275dc4481",
    "value": "sk_us-west-2_abcd1234efgh5678ijkl",
    "arn": "arn:aws:ivs:us-west-2:123456789:stream-key/62f15f1b-fe31-4127-b252-0666ac7f55a7",
    "tags": {}
  },
  "channel": {
    "name": "test-channel",
    "tags": {},
    "authorized": true,
    "latencyMode": "LOW",
    "ingestEndpoint": "jds34ksdg3las.global-contribute.live-video.net",
    "playbackUrl": "https://b37c565f6d79.us-west-2.playback.live-video.net/api/video/v1/aws.ivs.us-west-2.123456789.channel.oU40KS4LA1Dz.m3u8",
    "arn": "arn:aws:ivs:us-west-2:123456789:channel/fbc789c1-2c56-4ce6-a30a-d99275dc4481"
  }
}
```

要通过更新现有通道启用授权，请执行以下操作：

```
aws ivs update-channel --arn
arn:aws:ivs:us-west-2:693991300569:channel/742da049-fe9f-4f23-928e-c6753760a189
```

```
--authorized
```

这只是一个示例；您必须在 `--arn` 之后指定自己的通道 ARN。与创建通道时一样，在更新响应中，`authorized` 为 `true`。

## API 请求 ( 创建与更新 )

```
POST /CreateChannel HTTP/1.1
{
  "name": "<your channel name>",
  "authorized": true
}
```

```
POST /UpdateChannel HTTP/1.1
{
  "arn": "<channel arn>",
  "authorized": true
}
```

## 生成和签名播放令牌

有关使用 JWT 和受支持的库签名令牌的详细信息，请访问 [jwt.io](https://jwt.io)。在 `jwt.io` 界面上，您必须输入私有密钥才能签署令牌。只有当您想验证令牌时才需要公有密钥。

## 令牌架构

所有 JWT 都有三个字段：标头、有效负载和签名。

- 标头指定了如下内容：
  - `alg` 是签名算法。这是 ES384，是一种使用 SHA-384 哈希算法的 ECDSA 签名算法。
  - `typ` 是令牌类型 ( 即 JWT )。

```
{
  "alg": "ES384",
  "typ": "JWT"
}
```

- 有效负载包含特定于 Amazon IVS 的数据：
  - `channel-arn` 是视频播放请求的参考。

- `access-control-allow-origin` 是一个可选字段，可用于将播放限制到指定源；即设置为仅可从指定网站查看视频流。例如，您可能想要阻止用户将播放器嵌入到其他网站上。预设情况下，允许在所有源中播放。（请注意，这仅限制浏览器客户端；不会限制来自非浏览器客户端的播放。）此字段可以包含多个源，以逗号分隔。允许使用通配符域名：每个源的主机名都可以用 \* 开头（例如：`https://*.amazon.com`）。
- `strict-origin-enforcement` 是一个可选字段，可用于加强 `access-control-allow-origin` 字段中指定的源限制。默认情况下，`access-control-allow-origin` 限制仅适用于多变体播放列表。如果启用 `strict-origin-enforcement`，服务器将强制要求请求源与所有播放请求（包括多变体播放列表、变体播放列表和片段）的令牌匹配。这意味着所有客户端（包括非浏览器客户端）都必须在每个请求中提供有效的源请求标头。使用 `setOrigin` 方法在 IVS iOS 和 Android 回放器 SDK 中设置标头。除 iOS Safari 之外，它会在 Web 浏览器中自动设置。对于 iOS Safari，您需要将 `crossorigin="anonymous"` 添加到视频元素，以确保发送源请求标头。示例：`<video crossorigin="anonymous"></video>`。
- `single-use-uuid` 是可选字段，其中包含您在创作令牌时生成的有效的[通用唯一标识符 \(UUID\)](#)。如果添加此字段和 UUID 值，则生成的关联令牌在用于获取多变体播放列表和观看流后将失效。一次性身份验证令牌让恶意用户更难在您的私有通道上与其他观众分享流式传输。请注意，使用 `single-use-uuid` 申请时，`exp` 申请的最大值为未来 10 分钟。
- `viewer-id` 是一个可选字段，其中包含一个用于跟踪和引用授予令牌的查看者的 ID。此字段是必填字段，以便将来能够撤消查看者的查看会话。最大长度为 40 个字符，并且该值必须符合字符串的条件。请勿将此字段用于个人识别信息、机密信息或敏感信息。请注意，使用 `viewer-id` 时，`exp` 的最大值为未来 10 分钟。
- `viewer-session-version` 是一个可选字段，其中包含与此查看者会话关联的版本。当撤消查看者会话时，此值可用于筛选已撤消的查看者会话。例如，在此处指定 Unix 时间戳将允许撤消在指定时间之前启动的所有会话。该值必须是一个 64 位有符号整数 (Int64)。该字段应与 `viewer-id` 一起提供（可选）；它本身不起任何作用。默认值是 0。
- `exp` 是令牌到期时的 Unix UTC 时间戳。这并不代表视频流可以查看的时长。在观众初始化播放时验证令牌，而不是在整个视频流中。将该值作为整数类型值输入。

请注意，Unix 时间戳是一个数值，表示从 1970-01-01T00:00:00Z UTC 到指定 UTC 日期/时间的秒数（忽略闰秒）。不同的语言以不同的单位测量 Unix 时间戳；例如，JavaScript's `Date.now()` 以毫秒为单位返回时间。（请参阅 [JWT RFC 部分 4.1.4](#) 中的 `exp`。）

```
{
  "aws:channel-arn": "<channel_arn>",
  "aws:access-control-allow-origin": "<your-origin>",
  "aws:strict-origin-enforcement": true,
```

```
"aws:single-use-uuid": "<UUID>",  
"aws:viewer-id": "<viewer_id>",  
"aws:viewer-session-version": "<viewer_session_version>",  
"exp": <unix timestamp>  
}
```

- 要创建签名，可搭配使用私有密钥和标头 (ES384) 中指定的算法，对已编码的标头和已编码的负载进行签名。

```
ECDSASHA384(  
  base64UrlEncode(header) + "." +  
  base64UrlEncode(payload),  
  <private-key>  
)
```

## 说明

1. 使用 ES384 签名算法以及与其中一个播放密钥资源关联的私有密钥生成令牌签名（请参阅上面的 ECDSASHA384 示例）。
2. 汇编令牌。

```
base64UrlEncode(header) + "." +  
base64UrlEncode(payload) + "." +  
base64UrlEncode(signature)
```

3. 将签名令牌作为查询参数附加到播放 URL。

```
https://b37c565f6d790a14a0e78afaa6808a80.us-west-2.playback.live-video.net/  
api/video/v1/aws.ivs.us-west-2.123456789.  
channel.fbc789c1-2c56-4ce6-a30a-d99275dc4481.m3u8?token=<token>
```

## 列出播放密钥

Amazon IVS 客户可以随时获取其所有播放密钥资源的列表。

## 控制台说明

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。

2. 在左侧导航菜单中，选择“播放安全” > “播放键”。

系统将显示所有与您的账户关联的播放密钥资源。已删除的密钥不会显示，也没有过去密钥的历史记录。

## CLI 说明

```
aws ivs list-playback-key-pairs --region <aws-region>
```

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

响应示例：

```
{
  "keyPairs": [
    {
      "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad",
      "fingerprint": "81:f3:8c:88:78:61:4e:bc:58:07:a3:ca:63:f5:72:08",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3ff88c71-b18e-415f-948b-18bbde605a97",
      "fingerprint": "a2:b5:b3:0b:be:8e:73:00:0e:ad:e9:bb:02:c9:81:9a",
      "tags": {}
    }
  ]
}
```

## API 请求

有关用法信息，请参阅《IVS 低延迟流媒体 API 参考》[ListPlaybackKeyPairs](#)中的。

```
POST /ListPlaybackKeyPairs HTTP/1.1
{
  "maxResults": number,
  "nextToken": "string"
}
```



## 删除播放密钥

Amazon IVS 客户可以从其账户中删除播放密钥。删除的密钥将资源从客户账户中删除；使用已删除密钥签名的播放令牌将不会通过验证。

### 控制台说明

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在左侧导航菜单中，选择“播放安全” > “播放键”。
3. 选择要删除的密钥。
4. 选择 Delete (删除)。系统会显示 Delete playback key (删除播放密钥) 对话框。
5. 选择 Delete playback key (删除播放密钥)。

### CLI 说明

如果您拥有播放密钥的 ARN，则可以通过 Amazon CLI 删除该密钥。Amazon IVS 不支持通过 CLI 进行批量删除。

```
aws ivs delete-playback-key-pair --arn arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad --region <aws-region>
```

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

如果成功，则没有响应。您可以运行 `get` 命令（如下）来验证该密钥是否已删除。

以下是错误响应示例：

```
An error occurred (ResourceNotFoundException) when calling the DeletePlaybackKeyPair operation: ResourceNotFoundException:
```

### API 请求

```
POST /DeletePlaybackKeyPair HTTP/1.1
{
  "arn": "<playback key arn>"
}
```

## 获取有关播放密钥的信息

Amazon IVS 客户可以获取有关其播放密钥资源的信息。请务必注意，即使播放密钥是由 Amazon IVS 通过控制台创建的，关联的私钥也不可用。

### 控制台说明

1. 打开 [Amazon IVS 控制台](#)。如果您尚未选择通道所在的区域，请选择区域。
2. 在左侧导航菜单中，选择“播放安全” > “播放键”。
3. 选择要获取更多详细信息的密钥，然后选择 View details (查看详细信息)。

### CLI 说明

```
aws ivs get-playback-key-pair --arn arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad --region <aws-region>
```

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

响应示例：

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3ff88c71-b18e-415f-948b-18bbde605a97",
    "fingerprint": "a2:b5:b3:0b:be:8e:73:00:0e:ad:e9:bb:02:c9:81:9a",
    "tags": {}
  }
}
```

### API 请求

```
POST /GetPlaybackKeyPair HTTP/1.1
{
  "arn": "<playback key arn>"
}
```

## 撤消查看者会话

Amazon IVS 客户能够撤消与身份验证令牌关联的查看者会话，以阻止和停止使用该令牌进行播放。一个示例用例是将公有流转换为私有流，其中只有一部分公有流查看者可以继续观看。

有关以下说明中提及的 `viewer-id` 字段的信息，请参阅 [the section called “生成和签名播放令牌”](#) 下的“令牌架构”。

### CLI 说明

如果您有通道 ARN 和查看者 ID，则可以通过 AWS CLI 撤消查看者会话。

```
aws ivs start-viewer-session-revocation --channel-arn arn:aws:ivs:us-
west-2:991729659840:channel/abcdABCDefgh --viewer-id UDbh1u6M8nr0oarrzuKe --region
<aws-region>
```

可选输入 `--viewer-session-versions-less-than-or-equal-to <version>` 允许您指定一个筛选器，以便一次撤消查看者会话的哪个版本。

如果区域位于您的本地亚马逊云科技配置文件中，您可以忽略 `--region <aws-region>`。

如果成功，则没有响应。

以下是错误响应示例：

```
An error occurred (ValidationException) when calling the StartViewerSessionRevocation
operation: ValidationException:
```

### API 请求

```
POST /StartViewerSessionRevocation HTTP/1.1
{
  "channelArn": <channel ARN>,
  "viewerId": <viewer ID>,
  "viewerSessionVersionsLessThanOrEqualTo": <version>
}
```

还有一个 `BatchStartViewerSessionRevocation` 端点。请参阅 [IVS Low-Latency Streaming API Reference](#)。

## 自动录制到 Amazon S3 ( 低延迟直播功能 )

本节提供有关 Amazon IVS 低延迟流媒体的 auto-record-to-S3 功能的信息。我们讨论录制的 Amazon IVS 流的数据存储。并对存储内容和元数据文件架构予以说明。我们还讨论了录制内容的播放。

有关详细信息.....	请参阅 ...
设置和停止视频录制	Amazon IVS 入门中的 <a href="#">使用可选录制创建通道</a>
API	<a href="#">IVS API 参考</a>
成本	<a href="#">Amazon IVS 成本</a>

### S3 前缀

S3 前缀是所录制的每个实时流的唯一目录结构。实时流的所有媒体和元数据文件都写在此目录中。对于启用了录制功能的频道，S3 前缀是在直播会话开始时生成的，并将在录制开始和结束时在 CloudWatch 活动中提供。

S3 前缀采用以下格式：

```
/ivs/v1/<aws_account_id>/<channel_id>/<year>/<month>/<day>/<hours>/<minutes>/<recording_id>
```

其中：

- `aws_account_id` 是您的 AWS 账户的 ID ( 在您创建 AWS 账户时生成 )，通过该 ID 可创建通道。
- `channel_id` 是通道 ARN 的资源 ID 部分 ( Amazon Resource Name 的最后一部分 )。请参阅 [术语表](#) 中的 ARN。
- `<year>/<month>/<day>/<hours>/<minutes>` 是录制开始时的 UTC 时间戳。
- `recording_id` 是为每个录制会话生成的唯一 ID。

例如：

```
ivs/v1/123456789012/AsXego4U6tnj/2020/6/23/20/12/j8Z9091ndcVs
```

## 录制内容

录制开始时，视频段和元数据文件将写入为通道配置的 S3 存储桶。这些内容可用于后处理或作为按需视频播放。

请注意，直播开始并触发“录制开始”EventBridge 事件后，需要一段时间才能写入清单文件和视频片段。我们建议仅在发送“录制结束”事件后回放或处理录制的流。（请参阅 [将 Amazon EventBridge 与 IVS 配合使用](#)。）

以下是 Amazon IVS 实时会话录制的示例目录结构和内容：

```
ivs/v1/123456789012/AsXego4U6tnj/2020/6/23/20/12/j8Z9091ndcVs/  
  events  
    recording-started.json  
    recording-ended.json  
  media  
    hls  
    thumbnails
```

events 文件夹包含与录制事件相对应的元数据文件。记录开始、成功结束或以失败结束时会生成 JSON 元数据文件：

- events/recording-started.json
- events/recording-ended.json
- events/recording-failed.json

给定 events 文件夹将包含 recording-started.json 和 recording-ended.json 或 recording-failed.json 之一。

其中包含与录制会话及其输出格式相关的元数据。JSON 详细信息如下。

media 文件夹包含所有受支持的媒体内容，分为两个子文件夹：

- hls 包含实时会话期间生成的所有媒体和清单文件，并且可使用 Amazon IVS 播放器播放。此文件夹中有两种类型的 HLS 清单，即标准主清单 master.m3u8 和启用字节范围的清单 byte-range-multivariant.m3u8。因此，每个呈现文件夹都包含 playlist.m3u8 和一个 byte-range-variant.m3u8 文件。（请参阅下面的 [字节范围播放列表](#)。）
- thumbnails 包含实时会话期间生成的缩略图图像。缩略图每分钟生成并写入存储桶一次。（要更改此行为，请覆盖录制配置上的 thumbnailConfiguration 属性。）

**重要提示：**media 文件夹中的内容是动态生成并由第一次接收到的视频片的特征决定的；文件夹内容可能不代表最终特征（例如，呈现质量）。不要对静态路径做任何假设。要查找可用的 HLS 呈现及其路径，请使用以下所述的 JSON 元数据文件。

## 字节范围播放列表

除标准的 auto-record-to HLS 播放列表外，-S3 功能还支持[字节范围的播放列表](#)生成。字节范围播放列表符合第 4 版 HLS 规范。这样可以进行更精细的内容剪辑：在字节范围播放列表中，呈现索引文件中的每个片段引用视频块的字节子范围，提供比标准的 10 秒媒体文件大小更高的粒度。对于字节范围的播放列表，片段持续时间与为流配置的关键帧间隔相同。

## 缩略图

借助录制配置中的 thumbnailConfiguration 属性，您可以启用或禁用直播会话的缩略图录制，并修改了可为直播会话生成缩略图的时间间隔。缩略图间隔可以从 1 秒到 60 秒不等；默认情况下，缩略图录制处于启用状态，间隔为 60 秒。有关详细信息，请参阅 [Amazon IVS API 参考](#)。

缩略图配置还可能包括 storage 字段（SEQUENTIAL 和/或 LATEST）和分辨率指标（LOWEST\_RESOLUTION、SD、HD 或 FULL\_HD）。以下是每个选项的分辨率：

160 <= LOWEST\_RESOLUTION <= 360

360 < SD <= 480

480 < HD <= 720

720 < FULL\_HD <= 1080

## 合并片段化的流

录制配置的 recordingReconnectWindowSeconds 属性允许您指定一个时段（以秒为单位），在此期间，如果流中断并启动了新流，Amazon IVS 会尝试录制到与上一个流相同的 S3 前缀。换句话说，如果广播断开连接，然后在指定的间隔内重新连接，则多个流将被视为单个广播并合并在一起。

IVS 在 Amazon 中录制状态更改事件 EventBridge：录制结束事件和录制结束的 JSON 元数据文件至少会延迟，因 recordingReconnectWindowSeconds 为 Amazon IVS 会等待确保新的直播不会启动。

有关设置合并流功能的说明，请参阅《Amazon IVS 入门》中的[步骤 4：使用可选录制创建通道](#)。

## 资格

要将多个流录制到相同的 S3 前缀，所有流都必须满足某些条件：

- 视频的宽度和高度必须相同。
- 帧速率必须相同。
- 后续流的比特率之差必须小于或等于原始流比特率的 50%。
- 视频和音频编解码器必须相同。

备注：

- 最多合并 20 个流后，系统就会创建一个新的 S3 前缀。
- 48 小时后，系统将创建一个新的 S3 前缀。例如，如果第一个广播持续 48 小时，并在 `recordingReconnectWindowSeconds` 间隔内启动另一个广播，则下一个广播不会合并到第一个 S3 前缀中。
- 每个流必须在上一个流完成 10 秒或更长时间内开始。

## 已知问题

如果 `recordingReconnectWindowSeconds` 已启用并使用 Web 广播 SDK，则录制到相同的 S3 前缀可能不起作用，因为 Web 广播 SDK 会动态更改比特率和质量。

## JSON 元数据文件

当记录状态更改事件发生时，会生成相应的 Amazon CloudWatch 指标，并在 S3 前缀中写入元数据文件。（请参阅[监控 Amazon IVS 低延迟直播功能](#)。）

此元数据采用 JSON 格式，并包含以下信息：

字段	类型	必需	描述
<code>channel_arn</code>	字符串	是	广播实时流通道的 ARN。
<code>media</code>	对象	是	包含可用于此录制的媒体内容的枚举对象的对象。有效值："hls"、"thumbnails"。

字段	类型	必需	描述
hls	对象	是	描述 Apple HLS 格式输出的枚举字段。
duration_ms	integer	条件	所录制 HLS 内容的时长 ( 以毫秒为单位 )。此选项仅在 recording_status 为 "RECORDING_ENDED" 或 "RECORDING_ENDED_WITH_FAILURE" 时可用。如果在完成任何录制之前发生故障, 则该值为 0。
path	字符串	是	存储 HLS 内容的 S3 前缀的相对路径。
playlist	字符串	是	HLS 主播放列表文件的名称。
byte_range_playlist	字符串	是	HLS 字节范围多变体播放列表的名称。
renditions	对象	是	元数据对象的呈现数组 ( HLS 变体 )。始终至少有一个呈现。
path	字符串	是	为此呈现存储 HLS 内容的 S3 前缀的相对路径。
playlist	字符串	是	此呈现的媒体播放列表文件的名称。
byte_range_playlist	字符串	是	此呈现的字节范围播放列表的名称。
resolution_height	int	条件	编码视频的像素分辨率高度。仅当呈现包含视频轨道时, 此选项才可用。



字段	类型	必需	描述
<code>resolution_width</code>	int	条件	编码视频的像素分辨率宽度。仅当呈现包含视频轨道时，此选项才可用。
<code>thumbnails</code>	object	条件	描述缩略图输出的枚举字段。仅当缩略图配置的 <code>recordingMode</code> 为 <code>INTERVAL</code> 时才可用。
<code>path</code>	字符串	条件	存储缩略图内容的 S3 前缀的相对路径。仅当缩略图配置的 <code>recordingMode</code> 为 <code>INTERVAL</code> 时才可用。
<code>resolution_height</code>	int	是	缩略图的高度。默认值：源渲染的分辨率。该值受相关录制配置中的用户输入影响；具体来说，是 <code>thumbnailConfiguration.resolution</code> 值。
<code>resolution_width</code>	int	是	缩略图的宽度。默认值：源渲染的分辨率。该值受相关录制配置中的用户输入影响；具体来说，是 <code>thumbnailConfiguration.resolution</code> 值。
<code>latest thumbnail</code>	对象	是	描述最新缩略图输出的枚举字段。仅当缩略图配置的 <code>storage</code> 包含 <code>LATEST</code> 时才可用。
<code>resolution_height</code>	int	是	缩略图的高度。默认值为源渲染的分辨率。该值受相关录制配置中的用户输入影响；具体来说，是 <code>thumbnailConfiguration.resolution</code> 值。

字段	类型	必需	描述
<code>resolution_width</code>	int	是	缩略图的宽度。默认值为源渲染的分辨率。该值受相关录制配置中的用户输入影响；具体来说，是 <code>thumbnailConfiguration.resolution</code> 值。
<code>recording_ended_at</code>	字符串	条件	<p>录制结束时的 RFC 3339 UTC 时间戳。此选项仅在 <code>recording_status</code> 为 "RECORDING_ENDED" 或 "RECORDING_ENDED_WITH_FAILURE" 时可用。</p> <p><code>recording_started_at</code> 和 <code>recording_ended_at</code> 是这些事件生成时的时间戳，可能与 HLS 视频片段的时间戳不完全一致。要准确确定录制的持续时间，请使用 <code>duration_ms</code> 字段。</p>
<code>recording_started_at</code>	字符串	是	<p>录制开始时的 RFC 3339 UTC 时间戳。</p> <p>请参阅上面有关 <code>recording_ended_at</code> 的注释。</p>
<code>recording_status</code>	字符串	是	录制的状态。有效值："RECORDING_STARTED"、"RECORDING_ENDED"、"RECORDING_ENDED_WITH_FAILURE"。
<code>recording_status_message</code>	字符串	条件	状态的描述性信息。此选项仅在 <code>recording_status</code> 为 "RECORDING_ENDED" 或 "RECORDING_ENDED_WITH_FAILURE" 时可用。

字段	类型	必需	描述
version	字符串	是	元数据架构的版本。

## 示例：recording\_started.json

```
{
  "version" : "v1",
  "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
  "recording_started_at" : "2020-06-12T12:53:26Z",
  "recording_status" : "RECORDING_STARTED",
  "media" : {
    "hls" : {
      "path" : "media/hls",
      "playlist" : "master.m3u8",
      "byte_range_playlist": "byte-range-multivariant.m3u8",
      "renditions" : [
        {
          "path" : "480p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 480,
          "resolution_width" : 852
        },
        {
          "path" : "360p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 360,
          "resolution_width" : 640
        },
        {
          "path" : "160p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 160,
          "resolution_width" : 284
        },
        {
          "path" : "720p60",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
```

```
        "resolution_height" : 720,
        "resolution_width" : 1280
      }
    ]
  },
  "thumbnails": {
    "path": "media/thumbnails",
    "resolution_height": 480,
    "resolution_width": 852
  },
  "latest_thumbnail": {
    "path": "media/latest_thumbnail/thumb.jpg",
    "resolution_height": 480,
    "resolution_width": 852
  }
}
```

## 示例 : recording\_ended.json

```
{
  "version" : "v1",
  "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
  "recording_ended_at" : "2020-06-14T12:53:20Z",
  "recording_started_at" : "2020-06-12T12:53:26Z",
  "recording_status" : "RECORDING_ENDED",
  "media" : {
    "hls" : {
      "duration_ms" : 172794489,
      "path" : "media/hls",
      "playlist" : "master.m3u8",
      "byte_range_playlist": "byte-range-multivariant.m3u8",
      "renditions" : [
        {
          "path" : "480p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 480,
          "resolution_width" : 852
        },
        {
          "path" : "360p30",
          "playlist" : "playlist.m3u8",
```

```

        "byte_range_playlist": "byte-range-variant.m3u8",
        "resolution_height" : 360,
        "resolution_width" : 640
    },
    {
        "path" : "160p30",
        "playlist" : "playlist.m3u8",
        "byte_range_playlist": "byte-range-variant.m3u8",
        "resolution_height" : 160,
        "resolution_width" : 284
    },
    {
        "path" : "720p60",
        "playlist" : "playlist.m3u8",
        "byte_range_playlist": "byte-range-variant.m3u8",
        "resolution_height" : 720,
        "resolution_width" : 1280
    }
]
},
"thumbnails": {
    "path": "media/thumbnails",
    "resolution_height": 480,
    "resolution_width": 852
},
"latest_thumbnail": {
    "path": "media/latest_thumbnail/thumb.jpg",
    "resolution_height": 480,
    "resolution_width": 852
}
}
}

```

## 示例 : recording\_failed.json

```

{
    "version" : "v1",
    "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
    "recording_ended_at" : "2020-06-14T12:53:20Z",
    "recording_started_at" : "2020-06-12T12:53:26Z",
    "recording_status" : "RECORDING_ENDED",
    "media" : {
        "hls" : {

```

```
"duration_ms" : 172794489,
"path" : "media/hls",
"playlist" : "master.m3u8",
"byte_range_playlist": "byte-range-multivariant.m3u8",
"renditions" : [
  {
    "path" : "480p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 480,
    "resolution_width" : 852
  },
  {
    "path" : "360p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 360,
    "resolution_width" : 640
  },
  {
    "path" : "160p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 160,
    "resolution_width" : 284
  },
  {
    "path" : "720p60",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 720,
    "resolution_width" : 1280
  }
]
},
"thumbnails": {
  "path": "media/thumbnails",
  "resolution_height": 480,
  "resolution_width": 852
},
"latest_thumbnail": {
  "path": "media/latest_thumbnail/thumb.jpg",
  "resolution_height": 480,
  "resolution_width": 852
}
```

```
    }  
  }  
}
```

## 查找录制的呈现

当您内容流式传输到 Amazon IVS 频道时，`auto-record-to-s3` 会使用源视频生成多个格式副本。通过使用 [自适应比特率流](#) (ABR)，Amazon IVS 播放器会根据需要自动切换呈现（比特率），以优化不同网络条件下的播放效果。

在实时流式处理过程中生成的每个呈现都录制在 S3 录制前缀内的唯一路径中。在录制开始和停止期间，分辨率详细信息、路径和播放列表文件名会存储在 [JSON 元数据文件](#) 中。如果录制配置的 `renditionSelection` 值是 `ALL`，则选择所有渲染进行录制。如果 `renditionSelection` 为 `CUSTOM`，用户必须选择以下一个或多个选项：`LOWEST_RESOLUTION`、`SD`、`HD` 以及 `FULL_HD`。以下是每个选项的分辨率：

`160 <= LOWEST_RESOLUTION <= 360`

`360 < SD <= 480`

`480 < HD <= 720`

`720 < FULL_HD <= 1080`

**重要提示：**不要对静态呈现路径或生成的呈现列表做任何假设，因为这些信息可能会变化。不要假定特定呈现始终可用于 Amazon IVS 录制。要确定可用的呈现、分辨率和路径，请参阅元数据文件。

录制前缀中的 `event/recording_started.json` 或 `event/recording_ended.json` 文件包含录制前缀中媒体文件的路径和名称。所有 `path` 元素均相对于层次结构中的上一个路径。`media > hls` 下的元素描述了 HLS 资产，并在此级别定义了主播放列表名称和路径。

下面是一个 Python 代码片段，显示如何使用 S3 录制前缀和元数据文件生成主播放列表路径：

```
def get_master_playlist(metadata_json, s3_recording_prefix):  
    return s3_recording_prefix + '/' + metadata_json['media']['hls']['path'] + '/' +  
        metadata_json['media']['hls']['playlist']
```

`media > hls > renditions` 下的元素描述所录制呈现的列表。`resolution_height` 和 `resolution_width` 属性可用于确定视频分辨率。`path` 和 `playlist` 元素可用于派生呈现播放列表路径。使用这些字段可确定要使用哪个呈现进行任何后处理。

要发现录制文件的最高可用再现播放列表，您可以订阅“IVS 录制状态更改”事件。EventBridge（请参见 [将 Amazon EventBridge 与 IVS 配合使用](#)。）下面是一个示例 Python 脚本，演示如何使用订阅这些事件的 lambda 函数。

```
import json
import boto3
s3 = boto3.resource('s3')

def get_highest_rendition_playlist(bucket_name, prefix_name):
    object_path = "{}/events/recording-started.json".format(prefix_name)
    object = s3.Object(bucket_name, object_path)
    body = str(object.get()['Body'].read().decode('utf-8'))
    metadata = json.loads(body)
    media_path = metadata["media"]["hls"]["path"]
    renditions = metadata["media"]["hls"]["renditions"]

    highest_rendition = None
    highest_rendition_size = 0

    for rendition in renditions:
        current_rendition_size = rendition["resolution_height"]
        if (current_rendition_size > highest_rendition_size):
            highest_rendition_size = current_rendition_size
            highest_rendition = rendition

    highest_rendition_playlist = media_path + '/' + highest_rendition['path'] + '/' +
highest_rendition['playlist']
    return highest_rendition_playlist

def lambda_handler(event, context):
    prefix_name = event["detail"]["recording_s3_key_prefix"]
    bucket_name = event["detail"]["recording_s3_bucket_name"]
    rendition_playlist = get_highest_rendition_playlist(bucket_name, prefix_name)
    print("Highest rendition playlist: {}/{}".format(prefix_name, rendition_playlist))

    return {
        'statusCode': 200,
        'body': rendition_playlist
    }
```



## 播放私有存储桶中的录制内容

默认情况下，使用“自动录制到 Amazon S3”功能录制的对象为私有；因此，使用直接 S3 URL 无法访问这些对象。如果您尝试使用 Amazon IVS 播放器或其他播放器打开 HLS 主清单（m3u8 文件）进行播放，您将收到错误信息（例如，“您无权访问请求的资源”）。相反，您可以使用 Amazon CloudFront CDN（内容分发网络）播放这些文件。

### 亚马逊 CloudFront 配送

CloudFront 可以将发行版配置为提供来自私有存储桶的内容。通常，这比拥有可公开访问的存储桶更可取，在这种存储桶中，读取可以绕过提供的 CloudFront 控件。通过创建源访问控制 (OAC)，可以将您的分配设置为从私有存储桶提供服务，该访问控制是具有私有源存储桶读取权限的特殊 CloudFront 用户。您可以在创建发行版之后通过 CloudFront 控制台或 API 创建 OAC。请参阅[创建新的源访问控制](#)。

### 从亚马逊播放 CloudFront

使用 OAC 设置发行版以访问私有存储桶后，您的视频文件应该可以通过 CloudFront URL 进行使用。您的 CloudFront URL 是 AWS CloudFront 控制台详细信息选项卡上的分发域名。它应该如下所示：

```
a1b23cdef4ghij.cloudfront.net.
```

若要通过分配对录制的视频进行流式传输，请查找 master.m3u8 文件的对象键。它应该如下所示：

```
ivs/v1/012345678912/a0bCdEFGH1IjK/2021/4/20/12/03/aBcdEFghIjkl/media/hls/master.m3u8
```

将对象密钥附加到 CloudFront 网址的末尾。您的最终 URL 如下所示：

```
https://a1b23cdef4ghij.cloudfront.net/ivs/v1/012345678912/a0bCdEFGH1IjK/2021/4/20/12/03/aBcdEFghIjkl/media/hls/master.m3u8
```

要通过 Web 浏览器进行播放，请确保在两者 CloudFront 和 S3 存储桶中都配置 CORS。要进行 CloudFront 配置，请按照[创建源请求策略](#)中的说明将 CORS-S3 Origin 请求策略和 SimpleCors 响应标头策略附加到分配中 CloudFront。请参阅以下示例配置控制台页面：

## Cache key and origin requests

We recommend using a cache policy and origin request policy to control the cache key and origin requests.

Cache policy and origin request policy (recommended)

Legacy cache settings

### Cache policy

Choose an existing cache policy or create a new one.

CachingOptimized

Recommended for S3

Policy with caching enabled. Supports Gzip and Brotli compression.

[Create cache policy](#) [View policy](#)

### Origin request policy - optional

Choose an existing origin request policy or create a new one.

CORS-S3Origin

Policy for S3 origin with CORS

[Create origin request policy](#) [View policy](#)

### Response headers policy - optional

Choose an existing response headers policy or create a new one.

SimpleCORS

Allows all origins for simple CORS requests

[Create response headers policy](#) [View policy](#)

### ► Additional settings

有关 S3 CORS 配置，请参阅 [CORS 配置](#)，为您的 S3 存储桶创建相应的规则。

现在，您可以播放录制的视频，就像直接从存储桶播放一样。

有关更多信息，请参阅[限制对 Amazon S3 源的访问](#)。

## 将 Amazon EventBridge 与 IVS 低延迟直播配合使用

您可以使用亚马逊监控您的亚马逊 EventBridge 互动视频服务 (IVS) 直播。

Amazon IVS 会向亚马逊 EventBridge 发送有关您的直播状态的变更事件。传递的所有事件都有效。但是，事件将尽最大努力发出，这意味着并不能保证：

- 事件已传送 — 可能发生指定事件（例如，直播开始），但是 Amazon IVS 可能不会向其发送相应的变更事件。EventBridge Amazon IVS 尝试在放弃之前传递几个小时的事件。
- 事件将在指定的时间范围内传递 – 您可能会收到几个小时之前的事件。
- 按顺序传送事件：事件可能无序，尤其是在短时间内相互发送的情况下。例如，您会在先看到向下流，再看到向上流。

尽管事件丢失、延迟或无序的情况很少，但如果您编写了取决于通知事件的顺序或存在的关键业务程序，则应处理这些可能性。

您可以为以下任何事件创建 EventBridge 规则。

事件类型	活动	发送时间：
IVS 流状态更改	已创建会话	已成功使用了通道流密钥并创建了流会话。将在启动流时触发此事件，然后再处理视频或将视频传送给查看器。此事件可以帮助您确定流是否已启动但未能上线；例如，由于配置错误或违反限制。
IVS 流状态更改	会话已结束	编码器断开连接，并且 Amazon IVS 不再接收视频。此事件可以帮助您确定编码器何时停止发送媒体。  注意：当编码器断开连接时，“会话已结束”事件可能会发生在“流结束”事件之前。这是因为在“会话已结束”事件后的一小段时间内，Amazon IVS 仍在处理视频。

事件类型	活动	发送时间：
IVS 流状态更改	流启动	正在处理流，片段可供观众观看。此事件表示视频流正在处理中，查看器可以观看。此事件可以帮助您确定流是否成功上线。
IVS 流状态更改	流结束	流停止处理，不再为观众生成视频片段。此事件可以帮助您确定流何时结束，并且观众不能使用任何新的视频片段。（另请参阅“会话已结束”中的备注。）
IVS 流状态更改	流失败	由于超出了处理容量，流尚未处理且不可用。
IVS 流运行状况更改	饥饿开始	流没有从流播放器接收数据；流被称作处于“饥饿”状态。
IVS 流运行状况更改	饥饿结束	饥饿的流开始从流播放器接收数据，并且流再次运行正常。
IVS 限制违反	采集比特率	传入流的比特率超过 Amazon IVS 限制。
IVS 限制违反	采集分辨率	传入流的分辨率超过 Amazon IVS 限制。
IVS 限制违反	并发广播	同时流传输的通道总数超过 Amazon IVS 限制。
IVS 限制违反	并发观众	同时观看您通道的观众总数超过 Amazon IVS 限制。

事件类型	活动	发送时间：
IVS 录制状态更改	开始录制	开始处理流，并且已创建并验证录制前缀。将片段写入为通道配置的存储位置。  请注意，在实时流启动并发出录制开始事件后，需要一点时间将清单文件和视频段写入为通道配置的 S3 存储桶。我们建议仅在发送“录制结束”事件后回放或处理录制的流。
IVS 录制状态更改	结束录制	此通道的流结束并停止录制。
IVS 录制状态更改	录制启动失败	流启动，但由于错误而无法启动录制（例如，S3 存储桶不存在或不在正确的区域）。此实时流不会被录制。
IVS 录制状态更改	录制结束失败	由于录制过程中遇到错误，录制以失败结束（例如，如果尝试写入主播放列表失败）。某些对象可能仍会写入已配置的存储位置。

关于流 ID 的注释：stream\_id 字段（在许多事件中）是每次通道上线时分配的唯一流标识符。对于给定通道，每个实时流都有一个新的 stream\_id。因此，每个通道 ARN 可以有许多相应的流 ID。流 ID 允许客户区分同一通道上的不同流会话。

关于某些事件延迟的注释：编码器配置设置，特别是 IDR/关键帧间隔，会影响流开启的时间和相关事件（流开启和录制开启）的延迟。较短的关键帧间隔会减少此延迟。请参阅 Amazon IVS 流式处理配置中的[“减少延迟”](#)，了解有关设置 IDR/Keyframe 的信息。

## 为亚马逊 IVS 创建亚马逊 EventBridge 规则

您可以创建针对 Amazon IVS 发出的事件进行触发的规则。按照《亚马逊 EventBridge 用户指南》中的[“在亚马逊创建规则”](#) EventBridge 中的步骤进行操作。选择服务时，选择 Interactive Video Service (IVS)。

## 示例：流状态更改

流开始：当流正在处理且片段可供观众观看时，将发送此事件。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream Start",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

流结束：当流停止处理并且不再为观众生成视频片段时，将发送此事件。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream End",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

```
}
```

流失败：当流未处理且由于超出了处理容量而不可用时，将发送此事件。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream Failure",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
    "reason": "Transcode capacity exceeded. Please try again."
  }
}
```

## 示例：流运行状况更改

资源匮乏开始：当流未从流播放器接收数据时，将发送此事件；该流被认为处于“资源匮乏”状态。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream Health Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Starvation Start",
```

```

    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}

```

**资源匮乏结束**：当资源匮乏的流开始从流播放器接收数据并且流再次运行正常时，将发送此事件。

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream Health Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Starvation End",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}

```

## 示例：违反限制

所有违反限制事件都包括违反的限制的名称、限制值以及超出限制的数量（违反时的值减去限制）。

**摄取比特率**：当传入流的比特率超过 Amazon IVS 限制时，将发送此事件。

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [

```



```

    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "limit_name": "Ingest Bitrate",
    "limit_value": 1234,
    "exceeded_by": 3,
    "limit_unit": "bits per second",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}

```

摄取分辨率：当传入流的分辨率（像素总数或每个边缘的像素）超过 Amazon IVS 限制时，将发送此事件。

超出最大像素总数：

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "limit_name": "Ingest Resolution",
    "limit_value": 495000,
    "exceeded_by": 426600,
    "limit_unit": "total pixels",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}

```

超出每个边缘的最大像素数：

```
{
```

```

"version": "0",
"id": "01234567-0123-0123-0123-012345678901",
"detail-type": "IVS Limit Breach",
"source": "aws.ivs",
"account": "aws_account_id",
"time": "2017-06-12T10:23:43Z",
"region": "us-east-1",
"resources": [
  "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"TB
],
"detail": {
  "limit_name": "Ingest Resolution",
  "limit_value": 855,
  "exceeded_by": 45,
  "limit_unit": "pixels per edge",
  "channel_name": "Your Channel",
  "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
}
}

```

**并发广播**：当同时进行流传输的通道总数超过 Amazon IVS 限制时，将发送此事件。

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "limit_name": "Concurrent Broadcasts",
    "limit_value": 2,
    "exceeded_by": 3,
    "limit_unit": "active streams"
  }
}

```

**并发查看器**：当同时观看您通道的查看器总数超过 Amazon IVS 限制时，将发送此事件。

```

{

```

```
"version": "0",
"id": "01234567-0123-0123-0123-012345678901",
"detail-type": "IVS Limit Breach",
"source": "aws.ivs",
"account": "aws_account_id",
"time": "2017-06-12T10:23:43Z",
"region": "us-east-1",
"resources": [],
"detail": {
  "limit_name": "Concurrent Viewers",
  "limit_value": 10,
  "exceeded_by": 11,
  "limit_unit": "viewers"
}
}
```

## 示例：录制状态更改

对于所有录制状态更改事件，存储此直播所有对象的顶级路径为 `recording_s3_key_prefix`。在失败的情况下，失败的原因是 `recording_status_reason`。`recording_duration_ms` 字段是录制持续时间的毫秒数。

录制开始：当开始处理流并将片段写入为通道配置的存储位置时，将发送此事件。

```
{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",
  "account": "123456789012",
  "time": "2020-06-23T20:12:36Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"
  ],
  "detail": {
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
    "recording_status": "Recording Start",
    "recording_status_reason": "",
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
  }
}
```

```

    "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/
j8Z9091ndcVs",
    "recording_duration_ms": 0,
    "recording_session_id": "a6RfV23ES97iyfoQ"
  }
}

```

录制结束：当流结束并停止此通道的录制时，将发送此事件。

```

{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",
  "account": "123456789012",
  "time": "2020-06-24T07:51:32Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"
  ],
  "detail": {
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
    "recording_status": "Recording End",
    "recording_status_reason": "",
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
    "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/
j8Z9091ndcVs",
    "recording_duration_ms": 99370264,
    "recording_session_id": "a6RfV23ES97iyfoQ",
    "recording_session_stream_ids": ["st-254sopYUvi6F78ghp09vn0A",
"st-1A2b3c4D5e6F78ghij9Klmn"]
  }
}

```

录制开启失败：当流开启但由于错误（例如，S3 存储桶不存在或不在正确的区域）而无法开启时，将发送此事件。此实时流不会被录制。

```

{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",

```

```

"account": "123456789012",
"time": "2020-06-23T20:12:36Z",
"region": "us-west-2",
"resources": [
  "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"
],
"detail": {
  "channel_name": "Your Channel",
  "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
  "recording_status": "Recording Start Failure",
  "recording_status_reason": "ValidationException",
  "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
  "recording_s3_key_prefix": "",
  "recording_duration_ms": 0,
  "recording_session_id": "a6RfV23ES97iyfoQ"
}
}

```

录制结束失败：由于录制过程中遇到错误，录制结束失败时会发送此事件。某些对象可能仍会写入已配置的存储位置。

```

{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",
  "account": "123456789012",
  "time": "2020-06-24T07:51:32Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:ivs:us-west-2:123456a7-ab1c-2d34-e5f6-1a2b3c4d5678"
  ],
  "detail": {
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
    "recording_status": "Recording End Failure",
    "recording_status_reason": "InternalServerException",
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
    "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/j8Z9091ndcVs",
    "recording_duration_ms": 0,
    "recording_session_id": "a6RfV23ES97iyfoQ"
  }
}

```

```
}
```

## 使用 AWS CloudTrail 记录 Amazon IVS API 调用

Amazon Interactive Video Service (IVS) 与 AWS CloudTrail 集成，后者是一项提供 Amazon IVS 中由用户、角色或 AWS 服务所采取操作的记录的服务。CloudTrail 将 Amazon IVS 的所有 API 调用作为事件捕获。捕获的调用包括来自 Amazon IVS 控制台和来自您的应用程序的 API 调用。

如果您创建跟踪，则可以使 CloudTrail 事件持续传送到 Amazon S3 存储桶，包括 Amazon IVS 事件。如果您不配置跟踪，则仍可在 CloudTrail 控制台中的 Event history (事件历史记录) 中查看最新事件。使用 CloudTrail 收集的信息，您可以确定向 Amazon IVS 发出了什么请求、发出请求的 IP 地址、何人发出的请求、请求的发出时间以及其他详细信息。

要了解有关 CloudTrail 的更多信息，请参阅 [《AWS CloudTrail 用户指南》](#)。

### CloudTrail 中的 Amazon IVS 信息

在您创建账户时，对 AWS 账户启用了 CloudTrail。当 Amazon IVS 中发生活动时，该活动将记录在 CloudTrail 事件中，并与其他 AWS 服务事件一同保存在事件历史记录中。您可以在 AWS 账户中查看、搜索和下载最新事件。有关更多信息，请参阅[使用 CloudTrail 事件历史记录查看事件](#)。

要持续记录 AWS 账户中的事件（包括 Amazon IVS 的事件），请创建跟踪记录。通过 trail (跟踪)，CloudTrail 可将日志文件传送到 Amazon S3 存储桶。默认情况下，在 CloudTrail 控制台中创建跟踪记录时，此跟踪记录会应用于所有 AWS 区域。此跟踪记录来自 AWS 分区中的所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以分析在 CloudTrail 日志中收集的事件数据并采取操作。有关更多信息，请参阅 CloudTrail 用户指南中的这些项目：

- [为您的 AWS 账户创建跟踪记录](#) (概述)
- [CloudTrail 支持的服务和集成](#)
- [为 CloudTrail 配置 Amazon SNS 通知](#)
- [从多个区域接收 CloudTrail 日志文件](#)
- [从多个账户接收 CloudTrail 日志文件](#)

所有 Amazon IVS 操作均由 CloudTrail 记录，并记录在 [IVS Low-Latency Streaming API Reference](#)、[IVS Real-Time Streaming API Reference](#) 和 [IVS Chat API Reference](#) 中。例如，对 CreateChannel、ListChannels 和 DeleteChannel 终端节点的调用会在 CloudTrail 日志文件中生成条目。

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于确定发出的请求是否：

- 具有根凭证或 AWS Identity and Access Management (IAM) 用户凭证
- 使用角色或联合身份用户的临时安全凭证。
- 通过另一个 AWS 服务。

有关更多信息，请参阅 [CloudTrail userIdentity 元素](#)。

## 了解 Amazon IVS 日志文件条目

跟踪是一种配置，可用于将事件作为日志文件传送到您指定的 Amazon S3 存储桶。一个事件表示来自任何源的一个请求，包括有关所请求的操作、操作的日期和时间、请求参数等方面的信息。

CloudTrail 日志文件包含一个或多个日志条目。CloudTrail 日志文件不是公用 API 调用的有序堆栈跟踪，因此它们不会按任何特定顺序显示。

以下示例显示了用于 CreateChannel 终端节点的 CloudTrail 日志条目。

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ABCDEFGHIJK1L2EXAMPLE:account_name",
    "arn": "arn:aws:sts::123456789012:assumed-role/
First_Streamer/1234567890123456789",
    "accountId": "123456789012",
    "accessKeyId": "ABCDEFGHIJKL1EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ABCDEFGHIJK1L2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "First_Streamer"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-04-02T20:57:43Z"
      }
    }
  }
}
```



```
    },
    "eventTime": "2020-04-02T20:57:46Z",
    "eventSource": "ivs.amazonaws.com",
    "eventName": "CreateChannel",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "10.10.10.10",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
      "name": "default"
    },
  },
  "responseElements": {
    "channel": {
      "arn": "arn:aws:ivs:us-west-2:123456789012:channel/1EXAMPLE",
      "authorized": false,
      "ingestEndpoint": "EXAMPLE.global-contribute.live-video.net",
      "latencyMode": "LOW",
      "name": "default",
      "playbackUrl": "https://EXAMPLE.m3u8",
      "tags": {}
    },
    "streamKey": {
      "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/2EXAMPLE",
      "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/1EXAMPLE",
      "tags": {}
    }
  },
  "requestID": "12a34bc5-EXAMPLE",
  "eventID": "a1b2c3de-EXAMPLE",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

# Amazon IVS 安全性

AWS 的云安全性具有优先级最高。作为 AWS 客户，您将从专为满足大多数安全敏感型组织的要求而打造的数据中心和网络架构中受益。

安全性是AWS和您的共同责任。[责任共担模式](#)将其描述为云的安全性和云中的安全性：

- 云的安全性 – AWS 负责保护在 AWS 云中运行 AWS 服务的基础设施。AWS 还向您提供可安全使用的服务。作为[AWS 合规性计划](#)的一部分，第三方审核人员将定期测试和验证安全性的有效性。
- 云中的安全性 – 您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您组织的要求以及适用的法律法规。

此文档将帮助您了解如何在使用 Amazon IVS 时应用责任共担模式。以下主题说明如何配置 Amazon IVS 以实现您的安全性和合规性目标。

## 主题

- [数据保护](#)
- [身份和访问管理](#)
- [Amazon IVS 的托管式策略](#)
- [对 Amazon IVS 使用服务相关角色](#)
- [日志记录和监控](#)
- [事件响应](#)
- [恢复能力](#)
- [基础设施安全性](#)

## 数据保护

对于发送到 Amazon Interactive Video Service (IVS) 的数据，将应用以下数据保护措施：

- Amazon IVS 通过 HTTPS API 端点、RTMPS 提取和 HTTPS 播放对传输中数据进行加密。API 端点无需进行配置。
  - 在提取方面，流处理工具可以使用 RTMPS 保护内容的安全。默认情况下此功能可用。请参阅 [IVS 入门](#)。

- 可以将 IVS 通道配置为允许不安全的 RTMP 提取，但我们建议使用 RTMPS，除非是在需要 RTMP 的特定且经过验证的使用场景。
- 在转码/容器转换方面，可以在 Amazon 内部网络上以未加密的方式传输数据。
- 在播放方面，通过 HTTPS 提供数据。
- 实时视频内容是短暂存储，不是永久的。它只是通过系统传输，并在查看时缓存（在内部系统上）。
- 对于自动记录至 S3 功能，视频内容将写入 Amazon S3。有关更多信息，请参阅 [Amazon S3 中的数据保护](#)。
- 所有存储的客户输入元数据均位于使用服务器端加密的 AWS 托管式服务中。
- 为了提高服务质量，Amazon IVS 会存储客户（最终用户）元数据（例如，特定区域的缓冲速率）。此元数据不能用于供个人识别最终用户。
- 公有加密密钥（您管理的）可以与 `ImportPlaybackKeyPair` API 端点结合使用。请参阅 [IVS Low-Latency Streaming API Reference](#)。请勿共享这些加密密钥。

Amazon IVS 不会要求您提供任何客户（最终用户）数据。当需要您提供客户（最终用户）数据时，通道、输入或输入安全组中没有任何字段。

请勿将敏感的身份信息（如客户（最终用户）账号）放入自由格式字段（如 `Name`（名称）字段）。在使用 Amazon IVS 控制台或 API、AWS CLI 或 AWS SDK 的情况下也不例外。您输入到 Amazon IVS 中的任何数据都可能包含在诊断日志中。

流不是端到端加密的；流可以在 IVS 网络内部以不加密形式传输，以供处理。

## 身份和访问管理

AWS Identity and Access Management (IAM) 是一项 AWS 服务，可以帮助账户管理员安全地控制对 AWS 资源的访问。每个 AWS 资源都归某个 AWS 账户所有，创建和访问资源的权限由权限策略进行管理。IAM 账户管理员控制可以通过身份验证（登录）和授权（具有权限）使用 Amazon IVS 资源的用户。IAM 是为您的 AWS 账户提供的一项功能，不会额外收费。

**重要提示：**有关完整信息，请参阅 [AWS IAM 产品页面](#)、[IAM 用户指南](#) 以及 [签署 AWS API 请求](#)。在本节中，我们还提供了指向《IAM 用户指南》中特定章节的链接。您应在熟悉本材料后再继续使用。

## 受众

如何使用 IAM 因您可以在 Amazon IVS 中执行的操作而异。

- 服务用户 – 如果您使用 Amazon IVS 服务来完成工作，则管理员会为您提供所需的凭证和权限。随着您使用更多 Amazon IVS 功能来完成工作，您可能需要额外权限。了解如何管理访问权限可帮助您向管理员请求适合的权限。如果您无法访问 Amazon IVS 中的功能，请参阅 [故障排除](#)。
- 服务管理员 – 如果您在公司负责管理 Amazon IVS 资源，您可能对 Amazon IVS 具有完全访问权限。您有责任确定您的员工应访问哪些 Amazon IVS 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要了解有关您的公司如何将 IAM 与 Amazon IVS 搭配使用的更多信息，请参阅 [Amazon IVS 如何与 IAM 配合使用](#)。
- IAM 管理员 – 如果您是 IAM 管理员，您可以编写策略以管理对 Amazon IVS 的访问的详细信息。要查看您可在 IAM 中使用的 Amazon IVS 基于身份的策略示例，请参阅 [基于身份的策略示例](#)。

## Amazon IVS 如何与 IAM 配合使用

在您发出 Amazon IVS API 请求之前，您必须创建一个或多个 IAM 身份（用户、组和角色）和 IAM 策略，然后向身份附加策略。传播权限最多需要几分钟时间；在此之前，API 请求会被拒绝。

要简要了解 Amazon IVS 如何与 IAM 结合使用，请参阅《IAM 用户指南》中的 [与 IAM 配合使用的 AWS 服务](#)。

### 身份

您可以创建 IAM 身份，以便为向您的 AWS 账户中的人员和进程提供身份验证。IAM 组是 IAM 用户的集合，可以将其作为一个单位进行管理。请参阅《IAM 用户指南》中的 [身份（用户、组和角色）](#)。

### 策略

请参阅《IAM 用户指南》中的以下章节：

- [访问管理](#) – 关于策略的所有信息。
- [Amazon IVS 的操作、资源和条件键](#)
- [AWS 全局条件上下文键](#)
- [IAM JSON 策略元素参考](#) – 可以在 JSON 策略中使用的所有元素。

默认情况下，IAM 用户和角色没有创建或修改 Amazon IVS 资源（甚至更改自己的密码）的权限。他们也不能使用 AWS 控制台、AWS CLI 或 AWS API 执行任务。IAM 管理员必须创建 IAM 策略，以便为用户和角色授予权限以对所需的指定资源执行特定的 API 操作。

IAM 策略定义操作的权限，无关于您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。具有该策略的用户可以从 AWS 管理控制台、AWS CLI 或 AWS API 获取角色信息。

策略是由元素组成的 JSON 权限策略文档。Amazon IVS 支持三个元素：

- 操作 – Amazon IVS 的策略操作在操作前使用 `ivs` 前缀。例如，要授予某人使用 Amazon IVS `CreateChannel` API 方法创建 Amazon IVS 通道的权限，您应将 `ivs:CreateChannel` 操作纳入其策略中。策略语句必须包含 `Action` 或 `NotAction` 元素。
- 资源 – Amazon IVS 通道资源采用以下 [ARN](#) 格式：

```
arn:aws:ivs:${Region}:${Account}:channel/${channelId}
```

例如，要在语句中指定 `VgNkEJg0VX9N` 通道，请使用以下 ARN：

```
"Resource": "arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N"
```

无法对特定资源执行某些 Amazon IVS 操作，例如，用于创建资源的操作。在这些情况下，您必须使用通配符 (\*)：

```
"Resource": "*"
```

- 条件 – Amazon IVS 支持部分全局条件键：`aws:RequestTag`、`aws:TagKeys` 和 `aws:ResourceTag`。

在策略中，您可以使用变量作为占位符。例如，只有在使用用户的 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。请参阅 IAM 用户指南中的 [变量和标签](#)。

Amazon IVS 提供 AWS 托管式策略，可用于向身份授予一组预配置的权限（只读或完全访问权限）。您可以选择使用托管式策略，而不是下面所示的基于身份的策略。有关详细信息，请参阅 [Managed Policies for Amazon IVS](#)。

## 基于 Amazon IVS 标签的授权

您可以将标签附加到 Amazon IVS 资源，或者在请求中将标签传递给 Amazon IVS。要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。有关标记 Amazon IVS 资源的更多信息，请参阅 [IVS Low-Latency Streaming API Reference](#) 中的“Tagging”、[IVS Real-Time Streaming API Reference](#)，以及 [IVS Chat API Reference](#)。

有关示例，请参阅[基于标签查看 Amazon IVS 通道](#)。

## 角色

请参阅《IAM 用户指南》中的 [IAM 角色](#)和[临时安全凭证](#)。

IAM 角色是AWS账户中具有特定权限的实体。

Amazon IVS 支持使用临时安全凭证。您可以使用临时凭证进行联合身份登录，担任 IAM 角色或担任跨账户角色。您可以调用 [AWS Security Token Service](#) API 操作（如 AssumeRole 或 GetFederationToken）以获取临时安全凭证。

## 特权访问和非特权访问

API 资源具有特权访问权限。非特权播放访问权限可以通过专有通道进行设置；请参阅 [设置私有通道](#)。

## 使用策略的最佳实践

请参阅 [IAM 用户指南](#)中的 IAM 最佳实践。

基于身份的策略非常强大。它们确定某个人是否可以创建、访问或删除您账户中的 Amazon IVS 资源。这些操作可能会使 AWS 账户产生成本。请遵循以下建议：

- 授予最低权限 – 创建自定义策略时，仅授予执行任务所需的许可。最开始只授予最低权限，然后根据需要授予其他权限。这样做比起一开始就授予过于宽松的权限而后再尝试收紧权限来说更为安全。具体而言，预留 `ivs:*` 供管理员访问使用；请勿在应用程序中使用它。
- 为敏感操作启用多重身份验证 (MFA) – 为了提高安全性，要求 IAM 用户使用 MFA 访问敏感资源或 API 操作。
- 使用策略条件来增强安全性 – 在切实可行的范围内，定义基于身份的策略在哪些情况下允许访问资源。例如，您可编写条件来指定请求必须来自允许的 IP 地址范围。您也可以编写条件，以便仅允许指定日期或时间范围内的请求，或者要求使用 SSL 或 MFA。

## 基于身份的策略示例

### 使用 Amazon IVS 控制台

要访问 Amazon IVS 控制台，您必须具有一组最小权限，以允许您列出和查看有关 AWS 账户中 Amazon IVS 资源的详细信息。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该

策略的实体，控制台将无法按预期正常运行。为确保对 Amazon IVS 控制台的访问权限，请将以下策略附加到身份（参阅《IAM 用户指南》中的[添加和删除 IAM 权限](#)）。

以下策略的各个部分提供了对以下内容的访问权限：

- 所有 Amazon IVS API 终端节点
- 您的 Amazon IVS [服务限额](#)
- IVS 自动记录到 S3 功能所需的 Amazon S3 端点（低延迟直播功能）和 IVS 复合记录功能（实时直播功能）。
- 自动记录到 S3 服务链接角色创建
- Amazon Cloudwatch 用于获取实时流会话的指标

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "ivs:*",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
        "servicequotas:ListServiceQuotas"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
        "s3:CreateBucket",
        "s3:DeleteBucketPolicy",
        "s3:GetBucketLocation",
        "s3:GetBucketPolicy",
        "s3:ListAllMyBuckets",
        "s3:PutBucketPolicy"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
```

```

        "iam:AttachRolePolicy",
        "iam:CreateServiceLinkedRole",
        "iam:PutRolePolicy"
    ],
    "Effect": "Allow",
    "Resource": "arn:aws:iam::*:role/aws-service-role/ivs.amazonaws.com/
AWSServiceRoleForIVSRecordToS3*"
},
{
    "Action": [
        "cloudwatch:GetMetricData"
    ],
    "Effect": "Allow",
    "Resource": "*"
},
{
    "Action": [
        "lambda:AddPermission",
        "lambda:ListFunctions"
    ],
    "Effect": "Allow",
    "Resource": "*"
}
]
}

```

## 允许用户查看他们自己的权限

该示例显示了一项策略，该策略允许 IAM 用户查看附加到其用户身份的内联和托管策略。此策略包含在 AWS 控制台上或者使用 AWS CLI 或 AWS API 以编程方式完成此操作所需的权限。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ]
    }
  ]
}

```



```

    ],
    "Resource": [
        "arn:aws:iam:*:*:user/${aws:username}"
    ]
},
{
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

## 访问 Amazon IVS 通道

在本示例中，您想要为您 AWS 账户中的 IAM 用户授予访问其中一个 Amazon IVS 通道 VgNkEJgOVX9N 的权限。您还希望允许用户停止流 ( `ivs:StopStream` )、添加元数据 ( `ivs:PutMetadata` ) 和更新通道 ( `ivs:UpdateChannel` )。策略还授予了 Amazon IVS 控制台所需的权限： `ivs:ListChannels`、 `ivs:ListStreams`、 `ivs:GetChannel` 和 `ivs:GetStream`。

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ListChannelsInConsole",
            "Effect": "Allow",
            "Action": [
                "ivs:ListChannels",
                "ivs:ListStreams"
            ],
            "Resource": "arn:aws:ivs:*:*:channel/*"
        }
    ]
}

```

```

    },
    {
      "Sid": "ViewSpecificChannelInfo",
      "Effect": "Allow",
      "Action": [
        "ivs:GetChannel",
        "ivs:GetStream"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/VgNkJg0VX9N"
    },
    {
      "Sid": "ManageChannel",
      "Effect": "Allow",
      "Action": [
        "ivs:StopStream",
        "ivs:PutMetadata",
        "ivs:UpdateChannel"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/VgNkJg0VX9N"
    }
  ]
}

```

## 基于标签查看 Amazon IVS 通道

您可以在基于身份的策略中使用条件，以便基于标签控制对 Amazon IVS 资源的访问。此示例显示一项允许查看通道的策略。此策略还授予在 Amazon IVS 控制台上完成此操作的必要权限。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListWidgetsInConsole",
      "Effect": "Allow",
      "Action": "ivs:ListChannels",
      "Resource": "arn:aws:ivs:*:*:channel/*"
    },
    {
      "Sid": "ViewChannelIfOwner",
      "Effect": "Allow",
      "Action": "ivs:GetChannel",
      "Resource": "arn:aws:ivs:*:*:channel/*",
      "Condition": {

```

```
        "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
    }
}
]
```

您可以将该策略附加到您账户中的 IAM 用户。但是，仅当通道使用用户的用户名作为拥有者进行标记时，才授予此权限。如果用户 richard-roe 尝试查看 Amazon IVS 通道，则该通道必须标记为 Owner=richard-roe 或 owner=richard-roe；否则他会被拒绝访问。（条件标签键 Owner 匹配 Owner 和 owner，因为条件键名称不区分大小写。）

## 故障排除

可以使用以下信息，以帮助您诊断和修复在使用 Amazon IVS 和 IAM 时可能遇到的常见问题。

- 我无权在 Amazon IVS 中执行操作。

当 mateojackson IAM 用户尝试使用 AWS 控制台查看有关 API 的详细信息，但不具有 ivs:GetChannel 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
ivs:GetChannel on resource: arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N
```

在这种情况下，Mateo 请求他的管理员更新其策略，以允许他使用 ivs:GetChannel 操作访问 arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N 资源。

- 我想要查看我的访问密钥。

在创建 IAM 用户访问密钥后，您可以随时查看您的访问密钥 ID。但是，您无法再查看您的秘密访问密钥。如果您丢失了私有密钥，则必须创建一个新的访问密钥对。访问密钥分为两部分：

- 访问密钥 ID（例如 AKIAIOSFODNN7EXAMPLE）
- 秘密访问密钥（例如 wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY）

与用户名和密码一样，您必须同时使用访问密钥 ID 和秘密访问密钥对请求执行身份验证。像对用户名和密码一样，安全地管理访问密钥。

**重要提示：**请不要向第三方提供访问密钥，甚至为了帮助[找到您的规范用户 ID](#)也是如此。如果您这样做，可能会向某人提供对您的账户的永久访问权限。

当您创建访问密钥对时，系统会提示您将访问密钥 ID 和秘密访问密钥保存在一个安全位置。秘密访问密钥仅在您创建它时可用。如果丢失了您的秘密访问密钥，您必须为 IAM 用户添加新的访问密钥。

您最多可拥有两个访问密钥。如果您已有两个密钥，则必须删除一个密钥对，然后再创建新的密钥。请参阅 IAM 用户指南中的[管理 IAM 用户的访问密钥](#)。

- 我是管理员并希望允许其他人访问 Amazon IVS。

要允许其他人访问 Amazon IVS，您必须为需要访问权限的人员或应用程序创建一个 IAM 实体（用户或角色）。人员或应用程序将使用该实体的凭证访问 AWS。然后，您必须将策略附加到实体，以便在 Amazon IVS 中为他们（它们）授予正确的权限。

要开始使用，请参阅 IAM 用户指南中的[创建您的第一个 IAM 委派用户和组](#)。

- 我希望允许我的 AWS 账户以外的人访问我的 Amazon IVS 资源。

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表 (ACL) 的服务，您可以使用这些策略向人员授予对您的资源的访问权。如需了解相关信息，请参阅 IAM 用户指南：

如需了解 ...	请参阅 ...
如何跨您拥有的 AWS 账户提供对资源的访问权限	<a href="#">在您拥有的另一个 AWS 账户中向 IAM 用户提供访问权限</a>
如何向第三方 AWS 账户提供对您的资源的访问权限	<a href="#">向第三方拥有的 AWS 账户提供访问权限</a>
如何通过联合身份提供访问权限	<a href="#">向经过外部身份验证的用户（身份联合验证）提供访问权限</a>
使用角色和使用基于资源的策略进行跨账户访问之间的区别	<a href="#">IAM 角色与基于资源的策略有何不同</a>

## Amazon IVS 的托管式策略

AWS 托管策略是由 AWS 创建和管理的独立策略。AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住，AWS 托管策略可能不会为您的特定使用场景授予最低权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于使用场景的[客户管理型策略](#)来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新在 AWS 托管策略中定义的权限，则更新会影响该策略所附加到的所有主体身份（用户、组和角色）。当新的 AWS 服务启动或新的 API 操作可用于现有服务时，AWS 最有可能更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的[AWS 托管策略](#)。

## IVSReadOnlyAccess

使用 [IVSReadOnlyAccess](#) AWS 托管策略，您的应用程序开发人员可以访问所有非更改 IVS API 端点（低延迟和实时直播均适用）。

## IVSFullAccess

使用 [IVSFullAccess](#) AWS 托管策略，您的用户可以访问所有 IVS 和 IVS Chat API 端点（低延迟和实时直播均适用）。此策略包括对依赖服务的附加权限，以允许对 IVS 控制台的完全访问。

## 策略更新

查看自此服务开始跟踪 Amazon IVS 的 AWS 托管策略更新以来的更改详细信息。有关此页面更改的自动提示，请订阅 Amazon IVS 低延迟直播功能 [Document History](#) 页面上的 RSS 源。

更改	说明	日期
<a href="#">IVSFullAccess</a> – 新策略	IVS 添加了一项新策略，允许对 IVS（低延迟和实时直播）和 IVS 聊天功能进行完全访问。	2023 年 12 月 5 日

更改	说明	日期
<a href="#">IVSReadOnlyAccess</a> – 新策略	IVS 添加了一项新策略，允许对 IVS 进行只读访问（包括低延迟和实时直播）。	2023 年 12 月 5 日
Amazon IVS 已开启跟踪更改	Amazon IVS 为其 AWS 托管式策略开启了跟踪更改。	2023 年 12 月 5 日

## 对 Amazon IVS 使用服务相关角色

Amazon IVS 使用 IAM [服务相关角色](#)。服务相关角色是一种独特类型的 IAM 角色，它与 AWS 服务直接相关。服务相关角色是由 Amazon IVS 预定义的，并包含服务代表您调用其他 AWS 服务所需的所有权限。

服务相关角色可让您更轻松地进行设置 Amazon IVS，因为您不必手动添加必要的权限。Amazon IVS 定义了其服务相关角色的权限，并且仅 Amazon IVS 可以代入其角色。定义的权限包括信任策略和权限策略，以及不能附加到任何其它 IAM 实体的权限策略。

只有在先删除相关 IVS 资源后，才能删除 IVS 服务相关角色。这可以防止您无意中取消 IVS 访问与该服务相关角色相关的亚马逊云科技资源的权限。

有关支持服务关联的角色的其他服务的信息，请参阅[与 IAM 配合使用的亚马逊云科技服务](#)，并查找 Service-Linked Role（服务相关角色）列设为 Yes（是）的服务。选择 Yes（是），可转到查看该服务的[服务相关角色文档](#)的链接。

## Amazon IVS 的服务相关角色权限

Amazon IVS 使用名为 `AWSServiceRoleForIVSRecordToS3` 的服务相关角色代表您的 Amazon IVS 渠道访问 Amazon S3 存储桶。

`AWSServiceRoleForIVSRecordToS3` 服务相关角色信任以下服务来代入该角色：

- `ivs.amazonaws.com`

角色权限策略允许 Amazon IVS 对指定资源完成以下操作：

- 操作：`your Amazon S3 buckets` 上的 `s3:PutObject`

必须配置权限，允许 IAM 实体（如用户、组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅 IAM 用户指南中的[服务相关角色权限](#)。

## 为 Amazon IVS 创建服务相关角色

无需手动为 IVS 创建该服务相关角色。当您在 Amazon IVS 控制台、AWS CLI 或 AWS API 中创建录制配置资源时，Amazon IVS 将为您创建该角色。该服务相关角色命名为 `AWSServiceRoleForIVSRecordToS3`。

### Important

如果您在其他使用此角色支持的功能的服务中完成某个操作，此服务相关角色可以出现在您的账户中。要了解更多信息，请参阅[我的 IAM 账户中出现新角色](#)。

如果您删除了此服务相关角色然后需要再次创建它，则可以使用相同的流程在您的账户中重新创建此角色。当您创建录制配置资源时，Amazon IVS 将再次为您创建服务相关角色。

## 为 Amazon IVS 编辑服务相关角色

Amazon IVS 不允许您编辑 `AWSServiceRoleForIVSRecordToS3` 服务相关角色。创建服务相关角色后，您将无法更改角色的名称，因为可能有多种实体引用该角色。但是可以使用 IAM 编辑角色说明。有关更多信息，请参阅 IAM 用户指南中的[编辑服务相关角色](#)。

## 删除适用于 Amazon IVS 的服务相关角色

如果您不再需要使用某个需要服务相关角色的功能或服务，我们建议您删除该角色。这样您就没有未被主动监控或维护的未使用实体。但是，您必须先清除服务相关角色的资源，然后才能手动删除它。

### Note

如果在您试图删除资源时 Amazon IVS 服务正在使用该角色，则删除操作可能会失败。如果发生这种情况，则请等待几分钟后重试。

要删除 `AWSServiceRoleForIVSRecordToS3` 服务相关角色所使用的 Amazon IVS 资源：

使用 Amazon IVS 控制台、AWS CLI 或 AWS API 从所有渠道中删除录制配置关联，并删除该区域中的所有录制配置资源。

使用 IAM 手动删除服务相关角色：

使用 IAM 控制台、AWS CLI 或 AWS API 删除 `AWSServiceRoleForIVSRecordToS3` 服务相关角色。有关更多信息，请参阅 IAM 用户指南中的[删除服务相关角色](#)。

## Amazon IVS 服务相关角色支持的区域

Amazon IVS 支持在该服务可用的所有区域中使用服务相关角色。有关更多信息，请参阅[Amazon IVS 服务终端节点](#)。

## 日志记录和监控

要记录性能和/或操作，请使用 Amazon CloudTrail。请参阅[使用 AWS CloudTrail 记录 Amazon IVS API 调用](#)。

## 事件响应

要检测事件或发出事件警报，您可以通过 Amazon EventBridge 事件监控流的运行状况。请参阅适用于[低延迟直播功能](#)和适用于[实时直播功能](#)的 [Using Amazon EventBridge with Amazon IVS](#)。

使用[AWS Health Dashboard](#) 获取有关 Amazon IVS 整体运行状况的信息（按区域）。

## 恢复能力

IVS API 使用 AWS 全球基础设施，并围绕 AWS 区域和可用区构建。AWS 区域具有多个可用区，这些可用区实现了：

- 物理分离和隔离。
- 通过低延迟、高吞吐量、高度冗余的网络进行连接。
- 与传统的单个或多个数据中心基础设施相比，具有更高的可用性、容错性和可扩展性。

有关 API 的更多信息，请参阅[IVS Low-Latency Streaming API Reference](#)、[IVS Real-Time Streaming API Reference](#)，以及[IVS Chat API Reference](#)。有关 AWS 区域和可用区的更多信息，请参阅[AWS 全球基础设施](#)。



## Amazon IVS 视频数据层面

视频提取和分发通过 Amazon IVS 的内容交付网络 (CDN) 运行。CDN 专为低延迟视频设计，并进行了高度优化。这使得 Amazon IVS 能够以最低延迟为全球客户提供端到端的高品质视频。视频 CDN 具有全球入网点 (PoP)，可以覆盖地理分散的广播设备和查看器。

无论您选择在哪个 AWS 区域配置 Amazon IVS 资源：

- 流处理工具会自动将视频提取到它们所处位置附近的 POP。
- 查看器可以通过全球视频 CDN 流传输视频。

视频流提取完成后，将在多个 Amazon IVS 数据中心中的一个进行处理和转码。Amazon IVS 不会为提取或转码故障提供自动故障切换。而是流处理工具应对其编码器或广播客户端进行配置，以便在发生任何广播故障时自动重新提取。

## 基础设施安全性

作为一项托管式服务，Amazon IVS 都受 AWS 全球网络安全流程的保护。这些流程的说明详见[安全性、身份和合规性的最佳实践](#)。

## API 调用

您可以使用 AWS 发布的 API 调用通过网络访问 Amazon IVS。客户端必须支持传输层安全性协议 (TLS) 1.2 或更高版本。建议使用 TLS 1.3 或更高版本 (因为早期版本中存在漏洞)。客户端还必须支持具有完全向前保密 (PFS) 的密码套件，例如 Ephemeral Diffie-Hellman (DHE) 或 Elliptic Curve Ephemeral Diffie-Hellman (ECDHE)。大多数现代系统 (如 Java 7 及更高版本) 都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 委托人关联的秘密访问密钥来对 API 请求进行签名。您还可以使用[AWS Security Token Service](#) 生成临时安全凭证来对请求进行签名。

您可以从任何网络位置调用这些 API 操作，但 Amazon IVS 不支持基于资源的访问策略，其中可以包含基于源 IP 地址的限制。您还可以使用 Amazon IVS 策略来控制来自特定 Amazon Virtual Private Cloud (Amazon VPC) 端点或特定 VPC 的访问。事实上，这将对给定 Amazon IVS 资源的网络访问隔离在 AWS 网络中的特定 VPC。

此外，所有 API 请求均已签名 sigv4。

有关 API 的详细信息，请参阅[IVS Low-Latency Streaming API Reference](#)、[IVS Real-Time Streaming API Reference](#)，以及[IVS Chat API Reference](#)。

## 流式处理和播放

通过 HTTPS 进行从边缘到查看器的播放，如果通道配置为允许不安全的提取，则“贡献边缘”（提取端点）支持 RTMPS（RTMP over TLS）或 RTMP。Amazon IVS 流式处理需要使用 TLS 1.2 或更高版本。流不是端到端加密的；流可以在 IVS 网络内部以不加密形式传输，以供处理。

## 服务限额（低延迟流式传输）

以下是 Amazon Interactive Video Service (IVS) 端点、资源和其他操作的服务配额和限制。服务配额（也称为限制）是您的亚马逊云科技账户使用的服务资源或操作的最大数量。也就是说，除非表中另有说明，否则这些限制针对每个亚马逊云科技账户。另请参阅 [Amazon Service Quotas](#)。

要以编程方式连接到亚马逊云科技服务，您需要使用端点。另请参阅[亚马逊云科技服务端点](#)。

所有配额都是按区域强制执行的。

**重要提示：**所有账户都对并发视图和并发流数量有限制。（视图是一个独特的观看会话，它可以主动下载或播放视频。有关更详细的定义，请参阅 [术语表](#)。）确保您的限制足够，并在需要时请求增加限额，尤其是在您计划大型流传输活动时。

## 服务配额增加

对于可调配额，您可以通过 [Amazon 管理控制台](#) 请求提高速率。也可以使用控制台查看有关服务限额的信息。

API 调用速率限额不可调整。

## API 调用速率限额

端点类型	端点	默认
频道	BatchGetChannel	5 TPS
通道	CreateChannel	5 TPS
通道	DeleteChannel	5 TPS
通道	GetChannel	5 TPS
通道	ListChannels	5 TPS
通道	UpdateChannel	5 TPS
播放限制政策	CreatePlaybackRestrictionPolicy	5 TPS

端点类型	端点	默认
播放限制政策	DeletePlaybackRestrictionPolicy	5 TPS
播放限制政策	GetPlaybackRestrictionPolicy	5 TPS
播放限制政策	ListPlaybackRestrictionPolicies	5 TPS
播放限制政策	UpdatePlaybackRestrictionPolicy	5 TPS
私有通道	DeletePlaybackKeyPair	3 TPS
私有通道	GetPlaybackKeyPair	3 TPS
私有通道	ImportPlaybackKeyPair	3 TPS
私有通道	ListPlaybackKeyPairs	3 TPS
私有通道	BatchStartViewerSessionRevocation	2 TPS
私有通道	StartViewerSessionRevocation	10 TPS
录制配置	CreateRecordingConfiguration	3 TPS
录制配置	DeleteRecordingConfiguration	3 TPS
录制配置	GetRecordingConfiguration	3 TPS
录制配置	ListRecordingConfigurations	3 TPS
流	GetStream	5 TPS
流	GetStreamSession	5 TPS
流	ListStreams	5 TPS
流	ListStreamSessions	5 TPS

端点类型	端点	默认
流	PutMetadata	每个通道 5 TPS 每个账户 155 TPS
流	StopStream	5 TPS
流密钥	BatchGetStreamKey	5 TPS
流密钥	CreateStreamKey	5 TPS
流密钥	DeleteStreamKey	5 TPS
流密钥	GetStreamKey	5 TPS
流密钥	ListStreamKeys	5 TPS
标签	ListTagsForResource	10 TPS
标签	TagResource	10 TPS
标签	UntagResource	10 TPS

## 其他限额

资源或功能	默认	可调整	描述
通道	5000	是	每个 AWS 区域 的最大通道数。
并发流	100	是	每个 AWS 区域 可同时进行流传输的最大通道数量。如果超过此阈值，则流将被拒绝。
并发视图	15000	是	允许在 AWS 区域 的所有通道上播放实时通道的最大视图数

资源或功能	默认	可调整	描述
			。(视图是独特的观看会话，可自动下载视频或播放视频。请参阅此页开头的重要提示信息。)
采集比特率 (如果通道 type 为 BASIC)	1.5 Mbps 或 3.5 Mbps	否	<p>每秒可以流传输到 type 为 BASIC 的通道的最大比特数。</p> <ul style="list-style-type: none"> <li>如果输入视频质量为 480p 或更低，则默认限额为 1.5Mbps。</li> <li>如果输入视频质量高于 480p 但低于 1080p，则默认限额为 3.5Mbps。</li> </ul> <p>警告：如果超过此阈值，流可能会立即断开连接。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a>，了解有关通道 type 的详细信息。</p>
采集比特率 (如果通道 type 为 STANDARD)	8.5Mbps	否	<p>每秒可以流传输到 type 为 STANDARD (默认值) 通道的最大比特数。警告：如果超过此阈值，流可能会立即断开连接。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a>，了解有关通道 type 的详细信息。</p>

资源或功能	默认	可调整	描述
采集比特率 ( 如果通道 type 为 ADVANCED_HD )	8.5Mbps	否	每秒可以流传输到类型为 ADVANCED_HD 的通道的最大比特数。警告：如果超过此阈值，流可能会立即断开连接。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a> ，了解有关通道 type 的详细信息。
采集比特率 ( 如果通道 type 为 ADVANCED_SD )	8.5Mbps	否	每秒可以流传输到类型为 ADVANCED_SD 的通道的最大比特数。警告：如果超过此阈值，流可能会立即断开连接。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a> ，了解有关通道 type 的详细信息。
采集分辨率	1080p ( 总计 2100 万像素，1920 像素/边缘 )	否	可以流传输到通道的最大分辨率 ( 以像素为单位，不考虑其 type )。相关阈值有两个：像素总计和每个边缘的像素。警告：如果超过这些阈值，流可能会立即断开连接。请参阅 <a href="#">Amazon IVS Low-Latency Streaming API Reference</a> ，了解有关通道 type 的详细信息。
元数据负载	1KB	否	PutMetadata 请求有效负载 (Amazon IVS API) 的最大大小。
播放授权密钥对	3	否	每个 AWS 区域的最大播放授权密钥对数量。

资源或功能	默认	可调整	描述
播放限制政策	3	否	每个 AWS 区域的播放限制策略的最大数量。
播放限制政策国家	200	否	播放限制政策中 <code>allowedCountries</code> 列表的最大大小；即每项政策的最大国家/地区数。
播放限制政策的起源	5	否	播放限制策略中 <code>allowedOrigins</code> 列表的最大大小；也就是说，每个策略的最大来源数量。
播放限制策略来源长度	256	否	播放限制策略中 <code>allowedOrigins</code> 列表中条目的最大大小（以字符为单位）。
播放令牌大小	2KB	否	用于启动播放的整个 JSON Web 令牌 (JWT) 的最大大小。
录制配置	20	是	每个 AWS 区域的最大录制配置数量。
流密钥	1	否	每个通道的最大流密钥数。

## Service Quotas 与 CloudWatch 使用量指标集成

您可以使用通过 CloudWatch 使用量指标主动管理您的服务配额。您可以使用这些指标在 CloudWatch 图表和仪表板上可视化您当前的服务使用情况。Amazon IVS 用量指标与 Amazon IVS 服务配额对应。

您可以使用 CloudWatch 指标数学函数在图表上显示这些资源的服务配额。还可以配置警报，以在用量接近服务配额时向您发出警报。

要访问用量指标，请执行以下操作：

1. 打开服务配额控制台：<https://console.aws.amazon.com/servicequotas/>



2. 在导航窗格中，选择 AWS services ( AWS 服务 )。
3. 从 AWS 服务列表中，搜索并选择 Amazon Interactive Video Service。
4. 在服务配额列表中，选择感兴趣的服务配额。系统将打开一个新页面，其中包含有关服务配额/指标的信息。

或者，您可以通过 CloudWatch 控制台获取这些指标。在 AWS Namespaces ( AWS 命名空间 ) 中，选择 Usage ( 用量 )。然后，从 Service (服务) 列表中选择 IVS。( 请参阅 [监控 Amazon IVS 低延迟直播功能](#)。 )

在 AWS/Usage ( Amazon/用量 ) 命名空间中，Amazon IVS 将提供以下指标：

指标名称	描述
ResourceCount	您账户中运行的指定资源的数量。资源由与指标关联的维度定义。 有效统计数据：最大数量 ( 1 分钟期间内使用的最大资源数 )。

以下维度用于优化用量指标：

维度	描述
服务	包含资源的 AWS 服务的名称。有效值：IVS。
类	所跟踪的资源的类。有效值：None。
类型	所跟踪的资源类型。有效值：Resource。
资源	AWS 资源的名称。有效值：ConcurrentStreams、ConcurrentViews。 ConcurrentStreams 和 ConcurrentViews 使用量指标是 AWS/IVS 命名空间中指标的副本 ( 带有“无”维度 )，如监控 <a href="#">Amazon IVS 低延迟流媒体</a> 中所述。

## 为使用情况指标创建 CloudWatch 警报

要基于 Amazon IVS 使用量指标创建 CloudWatch 警报，请执行以下操作：

1. 从 Service Quotas 控制台中选择感兴趣的服务配额，如上所述。当前，只能为 ConcurrentStreams 和创建警报 ConcurrentViews。
2. 在 Amazon CloudWatch 警报部分，选择创建。
3. 从 Alarm threshold (警报阈值) 下拉清单中，选择要设置为警报值的已应用配额值的百分比。
4. 对于 Alarm name (警报名称)，输入警报的名称。
5. 选择 Create (创建)。

# Amazon IVS 流传输配置

开发人员使用 Amazon Interactive Video Service (IVS) 可轻松地面向全球观众提供低延迟的视频。借助 Amazon IVS，主播只需生成视频流，然后将视频流发送到 Amazon IVS 即可。Amazon IVS 使用 Amazon IVS 播放器处理视频（提取和转码），传送给观众并播放。

适用于实时流传输的解决方案有很多。无论您有一个工作室，配备了多台摄像头、视觉切换台、图形合成和各种混音设备，还是计划从智能手机上开始制作第一个视频流，都需要处理一些相同的概念和编码参数。

本文档介绍了如何配置视频编码器以流传输到 Amazon IVS。本文档的受众是希望在其应用程序中构建流传输功能的开发人员。

请注意，IVS 低延迟直播不支持纯音频输入。

## 先决条件

请按照 [IVS 入门](#) 中的步骤创建通道并设置流传输。在此过程中，会分配一个通道 ARN (Amazon Resource Name) 和流密钥，以及用于提取和播放流的 URL。您需要将流传输应用程序指向提取 URL。

阅读此文档之前，您应熟悉以下内容：

- Amazon IVS 基础知识：阅读 [What is IVS Low-Latency Streaming](#) 和 [IVS 入门](#)
- Amazon IVS API：了解 [IVS Low-Latency Streaming API Reference](#)。

## 降低延迟

Amazon IVS 低延迟流传输与大多数流传输应用程序兼容，只需对流传输应用程序配置进行少量更改。为了尽可能降低延迟，您必须使用 Amazon IVS 播放器；不支持第三方 HLS 视频播放器。请参阅 Amazon IVS 播放器开发工具包文档。

要为低延迟流传输准备流传输应用程序，请执行以下操作。（注意：并非所有这些选项都可用于每个流传输应用程序。）

- 在视频编码器上，设置 IDR/Keyframe 为 2 秒的间隔（或 1 秒，以获得更低的 end-to-end 延迟）。

IDR/Keyframe 直接影响直播启动的时间和相关 EventBridge 事件（直播开始和录制开始）的延迟。如果 IDR/Keyframe 为 2 秒，则流启动延迟将大约为 6-7 秒。如果 IDR/Keyframe 为 1 秒，

则流启动延迟将大约为 3-4 秒。仅在初始流启动延迟期之后，您的视频才可供观众观看，并自动录制到 Amazon S3。

更短的关键帧间隔 ( 1 秒 ) 存在 QoS 权衡。这可能会导致 Amazon IVS 播放器的自适应比特率流式处理 (ABR) 更频繁地切换分辨率；分段大小越小，因此 ABR 检查越频繁。由于提高了分辨率切换和/或观众的网络无法足够快速地下载分段，缓冲可能会增加。决定使用 1 秒还是 2 秒关键帧间隔时，评估这些权衡。

避免将 IDR/Keyframe 设置为高于 5 秒的值。直播流启动延迟不仅会比使用 1 或 2 秒时长，而且 IVS 也无法保证为播放生成的每个片段都以 IDR/keyframe 开头。当观看者开始播放或更改渲染时，不以 IDR/keyframe 开头的片段可能会导致解码错误或视觉失真。

- 如果适用，请在 x264 配置中将编码器设置为零延迟调整。
- 确保缓冲区大小 (Vbv) 不超过流的平均比特率 (kilobits-per-second)。

## 避免使用第三方流传输/转发服务

强烈建议不要使用第三方服务将内容重新流传输或转发到 Amazon IVS。这将导致额外的延迟。为了降低延迟，请直接流传输到 Amazon IVS。

## 编码器设置

### 流摄取：编解码器、RTMPS 和端口 443

编解码器：Amazon IVS 视频支持 H.264，音频支持 AAC (LC)。

Amazon IVS 支持流传输软件和硬件中最常用的安全提取协议 RTMPS ( 基于 TLS/SSL 连接的实时消息传输协议 )。Amazon IVS 流传输和播放需要使用 TLS 1.2 或更高版本。

您的视频编码器必须通过与出站端口 443/TCP 关联的 RTMPS 协议连接到 Amazon IVS 提取。为确保这一点，请指定 IVS 摄取服务器，其中包含路径中的端口：

```
rtmps://<IVS-ingest-server>/<IVS-stream-key>
```

例如：

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/<IVS-stream-key>
```

也可以将 IVS 通道配置为允许不安全的 RTMP 提取，但我们建议使用 RTMPS，除非是在需要 RTMP 的特定且经过验证的使用场景。流式传输 RTMP 时，务必将协议设置为 `rtmp://` 并删除 `:443` 端口。例如：

```
rtmp://a1b2c3d4e5f6.global-contribute.live-video.net/app/<IVS-stream-key>
```

## 分辨率/比特率/FPS

直播的分辨率在很大程度上决定了其比特率和帧速率 ( frames-per-second 或 FPS )。请使用以下建议准则。请注意，下面显示的分辨率是横向 ( 水平 x 垂直 )，反转即为纵向。

	可接受的质量 (SD) 480p (852x480)	优质 (HD) 720p (1280x720)	高品质 (全高清) 1080p (1920x1080)
比特率	高达 1500 Kbps	高达 4500 Kbps	高达 8500 Kbps
FPS	30	30 或 60	30 或 60
关键帧间隔	2 秒	2 秒	2 秒

比特率、FPS 和分辨率都是相互关联的。最佳值取决于具体情况，并且可能很复杂，因此难以确定。最佳建议是从上述值开始试验 ( 如需要 )。目标是在流传输过程中视频组件清晰流畅地运行，并在可用带宽内保持良好的分辨率。提高帧率和/或分辨率可提高整体视频质量，但这必然受带宽的限制。

Amazon IVS 可支持高达 60 FPS 的帧率 ( 包括欧洲 PAL 25 和 50 标准帧率 )。只要带宽足够，帧率越高，质量越好。根据应用的不同，低帧率也可以接受；例如，安全摄像头。

## 通道类型

通道类型决定可接受的分辨率和比特率。如果您超过支持的输入分辨率或比特率，则直播流可能会立即断开连接。

有四种通道类型：STANDARD、ADVANCED\_SD、ADVANCED\_HD，以及 BASIC。创建通道时，默认类型为 STANDARD。

根据通道类型，视频可能为已转码或已转码多路复用：

- STANDARD 和 ADVANCED 通道的视频已转码；从原始输入中生成多种品质，以根据观众的设备 and 网络情况自动为他们提供最佳体验。转码可在各种下载速度范围内实现更高的播放质量。
- BASIC 通道的视频已转码多路复用：Amazon IVS 将原始输入传送给查看器。

所有已转码通道都有已转码预设，用于确定生成哪些渲染。可以把它们想象成 ABR 阶梯。它们允许您在可用的下载带宽和视频质量之间进行权衡，以优化观看体验。

- STANDARD 通道有一个默认的转码预设。
- ADVANCED 通道有两个可选的转码预设：
  - 受限的带宽传输对每个质量级别使用比 STANDARD 低的比特率。如果您的下载带宽较低和/或视频内容简单（例如，讲话者头部特写），则请使用它。
  - 较高的带宽传输对每个质量级别使用较高的比特率。如果您有较高的下载带宽和/或复杂的视频内容（例如闪光灯和快速场景切换），则请使用它。这是默认模式。

## 标准通道

STANDARD 通道已转码。生成的最高视频分辨率为全高清，1080p。这是默认通道类型。

- 转码预设：有一个默认的转码预设阶梯。
- 音频：对于 360p 及以下的渲染，音频会进行转码。对于其他渲染，将传递原始音频。

输入分辨率和最大比特率	阶梯详情
在 8.5Mbps 下为 1080p 60 帧	<ol style="list-style-type: none"> <li>1. 在源码率下为 1080p 60 帧</li> <li>2. 在 3.4Mbps 下为 720p 60 帧</li> <li>3. 在 1.4Mbps 下为 480p 30 帧</li> <li>4. 在 0.63Mbps 下为 360p 30 帧</li> <li>5. 在 0.23Mbps 下为 160p 30 帧</li> </ol>
在 8.5Mbps 下为 1080p 30 帧	<ol style="list-style-type: none"> <li>1. 在源码率下为 1080p 30 帧</li> <li>2. 在 2.4Mbps 下为 720p 30 帧</li> <li>3. 在 1.4Mbps 下为 480p 30 帧</li> <li>4. 在 0.63Mbps 下为 360p 30 帧</li> <li>5. 在 0.23Mbps 下为 160p 30 帧</li> </ol>
在 8.5Mbps 下为低于 1080p 60 帧且大于 720p 60 帧	<ol style="list-style-type: none"> <li>1. 源传递</li> <li>2. 在 3.4Mbps 下为 720p 60 帧</li> <li>3. 在 1.4Mbps 下为 480p 30 帧</li> <li>4. 在 0.63Mbps 下为 360p 30 帧</li> <li>5. 在 0.23Mbps 下为 160p 30 帧</li> </ol>

输入分辨率和最大比特率	阶梯详情
在 8.5Mbps 下为低于 1080p 30 帧且大于 720p 30 帧	<ol style="list-style-type: none"> <li>1. 源传递</li> <li>2. 在 2.4Mbps 下为 720p 30 帧</li> <li>3. 在 1.4Mbps 下为 480p 30 帧</li> <li>4. 在 0.63Mbps 下为 360p 30 帧</li> <li>5. 在 0.23Mbps 下为 160p 30 帧</li> </ol>
在 8.5Mbps 下为 720p 60 帧	<ol style="list-style-type: none"> <li>1. 在 3.4Mbps 下为 720p 60 帧</li> <li>2. 在 1.4Mbps 下为 480p 30 帧</li> <li>3. 在 0.63Mbps 下为 360p 30 帧</li> <li>4. 在 0.23Mbps 下为 160p 30 帧</li> </ol>
在 8.5Mbps 下为 720p 30 帧	<ol style="list-style-type: none"> <li>1. 在 2.4Mbps 下为 720p 30 帧</li> <li>2. 在 1.4Mbps 下为 480p 30 帧</li> <li>3. 在 0.63Mbps 下为 360p 30 帧</li> <li>4. 在 0.23Mbps 下为 160p 30 帧</li> </ol>
在 8.5Mbps 下为小于 720p 30 帧/60 帧且大于或等于 480p 30 帧/60 帧	<ol style="list-style-type: none"> <li>1. 在 1.4Mbps 下为 480p 30 帧</li> <li>2. 在 0.63Mbps 下为 360p 30 帧</li> <li>3. 在 0.23Mbps 下为 160p 30 帧</li> </ol>

## 高级高清通道

ADVANCED-HD 通道已转码。制作的最高视频分辨率为高清，720p。

- 转码预设：有两个可选择的转码预设阶梯。
- 音频：对于 360p 及以下的渲染，音频会进行转码。对于其他渲染，将传递原始音频。

输入分辨率和最大比特率	阶梯详情
在 8.5Mbps 下为 720p 60 帧至 1080p 60 帧	<p>转码预设：更高的带宽传输（默认）：</p> <ol style="list-style-type: none"> <li>1. 在 3Mbps 下为 720p 60 帧</li> <li>2. 在 1.3Mbps 下为 480p 30 帧</li> <li>3. 在 0.7Mbps 下为 360p 30 帧</li> <li>4. 在 0.27Mbps 下为 160p 30 帧</li> </ol>

输入分辨率和最大比特率	阶梯详情
	<p>5. 在 0.08Mbps 下仅限音频</p> <p>转码预设：带宽传输受限：</p> <ol style="list-style-type: none"> <li>1. 在 2.2Mbps 下为 720p 60 帧</li> <li>2. 在 0.8Mbps 下为 480p 30 帧</li> <li>3. 在 0.4Mbps 下为 360p 30 帧</li> <li>4. 在 0.22Mbps 下为 160p 30 帧</li> <li>5. 在 0.08Mbps 下仅限音频</li> </ol>
在 8.5Mbps 下为 720p 30 帧至 1080p 30 帧	<p>转码预设：更高的带宽传输（默认）：</p> <ol style="list-style-type: none"> <li>1. 在 2.3Mbps 下为 720p 30 帧</li> <li>2. 在 1.3Mbps 下为 480p 30 帧</li> <li>3. 在 0.7Mbps 下为 360p 30 帧</li> <li>4. 在 0.27Mbps 下为 160p 30 帧</li> <li>5. 在 0.08Mbps 下仅限音频</li> </ol> <p>转码预设：带宽传输受限：</p> <ol style="list-style-type: none"> <li>1. 在 1.9Mbps 下为 720p 30 帧</li> <li>2. 在 0.8Mbps 下为 480p 30 帧</li> <li>3. 在 0.4Mbps 下为 360p 30 帧</li> <li>4. 在 0.22Mbps 下为 160p 30 帧</li> <li>5. 在 0.08Mbps 下仅限音频</li> </ol>



输入分辨率和最大比特率	阶梯详情
<p>在 8.5Mbps 下为低于 720p 30 帧/60 帧且大于 480p 30 帧/60 帧</p>	<p>转码预设：更高的带宽传输（默认）：</p> <ol style="list-style-type: none"> <li>1. 以 2.3Mbps 的速度进行源转码</li> <li>2. 在 1.3Mbps 下为 480p 30 帧</li> <li>3. 在 0.7Mbps 下为 360p 30 帧</li> <li>4. 在 0.27Mbps 下为 160p 30 帧</li> <li>5. 在 0.08Mbps 下仅限音频</li> </ol> <p>转码预设：带宽传输受限：</p> <ol style="list-style-type: none"> <li>1. 以 1.9Mbps 的速度进行源转码</li> <li>2. 在 0.8Mbps 下为 480p 30 帧</li> <li>3. 在 0.4Mbps 下为 360p 30 帧</li> <li>4. 在 0.22Mbps 下为 160p 30 帧</li> <li>5. 在 0.08Mbps 下仅限音频</li> </ol>
<p>在 8.5Mbps 下为 480p 30 帧/60 帧</p>	<p>转码预设：更高的带宽传输（默认）：</p> <ol style="list-style-type: none"> <li>1. 在 1.3Mbps 下为 480p 30 帧</li> <li>2. 在 0.7Mbps 下为 360p 30 帧</li> <li>3. 在 0.27Mbps 下为 160p 30 帧</li> <li>4. 在 0.08Mbps 下仅限音频</li> </ol> <p>转码预设：带宽传输受限：</p> <ol style="list-style-type: none"> <li>1. 在 0.8Mbps 下为 480p 30 帧</li> <li>2. 在 0.4Mbps 下为 360p 30 帧</li> <li>3. 在 0.22Mbps 下为 160p 30 帧</li> <li>4. 在 0.08Mbps 下仅限音频</li> </ol>

## 高级标清通道

ADVANCED-SD 通道已转码。可用的渲染以输入质量为上限，没有向上转换。

- 转码预设：有两个可选择的转码预设阶梯。

- 音频：音频已转码。

输入分辨率和最大比特率	阶梯详情
在 8.5Mbps 下为 480p 30 帧/60 帧至 1080p 30 帧/60 帧	转码预设：更高的带宽传输（默认）： <ol style="list-style-type: none"> <li>1. 在 1.3Mbps 下为 480p 30 帧</li> <li>2. 在 0.7Mbps 下为 360p 30 帧</li> <li>3. 在 0.27Mbps 下为 160p 30 帧</li> <li>4. 在 0.08Mbps 下仅限音频</li> </ol> 转码预设：带宽传输受限： <ol style="list-style-type: none"> <li>1. 在 0.8Mbps 下为 480p 30 帧</li> <li>2. 在 0.4Mbps 下为 360p 30 帧</li> <li>3. 在 0.22Mbps 下为 160p 30 帧</li> <li>4. 在 0.08Mbps 下仅限音频</li> </ol>

## 基本通道

BASIC 通道已转码多路复用。生成了一份渲染。

- 转码预设：不适用
- 音频：音频已转码。

输入分辨率和最大比特率	阶梯详情
大于 480p 30 帧/60 帧且小于或等于 1080p 30 帧/60 帧，带宽为 3.5Mbps	源编码参数（无阶梯）
在 1.5Mbps 下为 480p 30 帧/60 帧	源编码参数（无阶梯）

## 视频设置

我们建议使用以下设置。这些设置适用于大多数 H.264 视频编码软件或硬件 API。

- 在视频编码器上，设置IDR/Keyframe为 2 秒的间隔（或 1 秒，以获得更低的 end-to-end 延迟）。
- H.264 级别：主要
- 场景变化：关闭（首选）
- 色度子样本：YUV420P
- CABAC：首选
- ColorSpace: BT.709（为了最大限度地提高高清电视和电脑显示器的兼容性，建议使用）。Amazon IVS 视频转码支持 ColorSpace直通；高级用户可以使用其他 ColorSpace 视频和全方位视频。

## 音频设置

我们支持以下设置：

- 编解码器：AAC (LC)
- 比特率：96 Kbps 至 320 Kbps
- 采样率：44.1 Khz 或 48 Khz（最好与您的生产音频流相匹配）
- 通道：最大 2 个 – 立体声（支持 1：单声道或 2：立体声音频通道）

## 使用 CBR，而不是 VBR

始终使用 CBR（常量 BitRate），而不是 VBR（变量 BitRate）作为编码器的速率控制方法。CBR 更适用于固定带宽特性的网络，为客户端设备提供更可预测、更稳定的视频播放。使用恒定比特率，观众可轻松选择其连接随着时间的推移能够处理的质量级别。

根据场景的复杂程度，VBR 可能会导致比特率峰值，这可能会导致视频到达 Amazon IVS 之前帧丢弃和/或在客户端播放器中进行缓冲。

我们强烈建议只使用 CBR。如果您使用 VBR，流将更容易受到缓冲和播放不流畅的影响。

## 使用渐进式信号

使用渐进式信号流；在制作流程和/或编码中避免任何隔行视频。渐进式流信号可生成更好的播放质量，一次显示整个帧，可避免在显示隔行信号时产生的任何运动失真。

## 网络要求

须拥有稳定的互联网连接，保持足够的、持续的上传流。不稳定的互联网连接可能会导致观众的视频流卡顿和滞后。

使用有线连接。WiFi 而且，由于 QoS/Packet-Queue 优先级不佳，LTE 连接可能参差不齐或受到干扰或延迟。尽可能使用硬连线的连接来传输视频流。

计划分配的带宽比所需的最小带宽多 50%。多出的带宽是为了补偿视频比特流编码时的比特率波动。

使用专用的互联网 VLAN 对计算机进行编码。编码器使用单独的网络，可防止潜在的破坏性影响，包括：流量污染、带宽瓶颈和不利的安全因素。

## 隐藏式字幕

IVS 支持隐藏式字幕。如果主播想要为观众提供字幕，必须通过视频编码器以接受的格式传输字幕数据，要么将其嵌入到视频流中，或者与视频流一起传输。

Amazon IVS 接受线路 21 CEA-708/EIA-608 格式的字幕（也称为 608 over 708）。您可以使用以下方法之一传输字幕：

- CEA-708/EIA-608 嵌入到基础视频流中，如 ATSC A/72 (SEI user\_data) 中所述。这种格式在电视广播编码器中很常见。
- CEA-708/EIA-608 通过 RTMPS 脚本/amf0 标签传输。onCaptionInfo 这种格式在互联网广播编码器和媒体服务器（如 Elemental Technologies 和 Wowza）中很常见。Amazon IVS 播放器软件开发工具包仅支持一种语言；不支持多轨道字幕播放。

注意：Amazon IVS 播放器软件开发工具包仅在 CC1 NTSC 字段 1 中支持字幕数据。不支持多轨道字幕回放。

通过 RTMPS 传输时，有效负载须包含一个具有两个元素对的 ECMA 数组：

- 名为 type 的字符串，包含字符 708。
- 名为 data 的字符串，其中包含以 base64 编码的 CEA-708/EIA-608 有效负载。

例如：

```
00000000 12 00 00 69 00 00 00 00 00 00 00 02 00 0d 6f 6e |...i.....on|
```

```

00000010 43 61 70 74 69 6f 6e 49 6e 66 6f 08 00 00 02 |CaptionInfo.....|
00000020 00 04 74 79 70 65 02 00 03 37 30 38 00 04 64 61 |..type...708..da|
00000030 74 61 02 00 3c 74 51 41 78 52 30 45 35 4e 41 4e |ta..<tQAxR0E5NAN|
00000040 4c 41 50 79 55 72 76 79 55 49 50 79 52 51 50 7a |LAPyUrvyUIPyRQPz|
00000050 49 35 66 7a 73 37 50 7a 76 4c 50 77 67 56 50 7a |I5fzs7PzvLPwgVPz|
00000060 33 36 66 7a 30 34 2f 78 6f 67 50 79 55 4c 2f 38 |36fz04/xogPyUL/8|
00000070 3d 00 00 09 00 00 00 74 |=......t|

```

如果使用 Elemental 视频编码器，请按如下方式进行设置：

- 将字幕嵌入设置为“capture 608 Field 1”。
- 在输出组中嵌入字幕onCaptionInfo作为 RTMPS 标签。

有关更多信息，请参阅博客文章[添加隐藏式字幕到 Amazon IVS 实时流](#)。

## 使用 FFmpeg 的流

FFmpeg 是一个免费的开源项目，包括一个庞大的软件库，用于处理视频、音频以及其他多媒体文件和流。它可与许多操作系统和设备配合使用。

请参阅 [FFmpeg 网站](#)，了解如何安装 FFmpeg 以及其他相关信息。使用最新的静态构建（不编译）。

安装后，为 FFmpeg 选择音频/视频输入源。您可以查找可用的内容，如下所示：

```
ffmpeg -list_devices true -f dshow -i dummy.
```

有关更多信息，请单击[此处](#)。根据可用的内容和针对的捕获方法，您应该能够直接从选定的设备捕获视频/音频（嵌入式），并使用 FFmpeg 对信号进行编码。例如：

- 网络摄像头 – 要捕获 Logitech C920 网络摄像头的输出，请执行以下操作：

```

ffmpeg -f dshow -video_size 1920x1080 -framerate 30 -i video="HD Pro Webcam
C920":audio="Microphone (HD Pro Webcam C920)" -c:v libx264 -b:v 6000K -maxrate
6000K -pix_fmt yuv420p -r 30 -s 1920x1080 -profile:v main -preset veryfast -g 120
-x264opts "nal-hrd=cbr:no-scenecut" -acodec aac -ab 160k -ar 44100 -f flv rtmps://
<IVS-ingest-server>/<IVS-stream-key>

```

- 视频文件 – FFmpeg 适用于多种视频文件格式和捕捉卡。以下是基于 MP4 输入的流传输示例：

```

ffmpeg -re -i input.mp4 -c:v libx264 -b:v 6000K -maxrate 6000K -pix_fmt yuv420p -s
1920x1080 -profile:v main -preset veryfast -force_key_frames expr:gte(t,n_forced*2)

```

```
-x264opts "nal-hrd=cbr:no-scenecut" -acodec aac -ab 160k -ar 44100 -f flv rtmps://<IVS-ingest-server>/app/<IVS-stream-key>
```

有关 <IVS-ingest-server> 和 <IVS-stream-key> 输入内容的更多信息，请参阅 [IVS 入门](#) 中有关设置实时流传输的信息。例如：

- 提取服务器：rtmps://jds34ksdg3las.global-contribute.live-video.net/app/
- 流密钥：sk\_us-west-2\_abcd1234efgh5678ijkl

## 使用 Amazon IVS 广播 SDK 的流

Amazon IVS 广播 SDK 适用于使用 Amazon IVS 构建 Android、iOS 或 Web 应用程序的开发人员。请参阅《Amazon IVS 用户指南》中的广播开发工具包文档，在[此处](#)了解如何开始。包含适用于 Android、iOS 和 Web 流式传输指南的子页面。您可以使用广播 SDK 自定义比特率、帧率和分辨率。

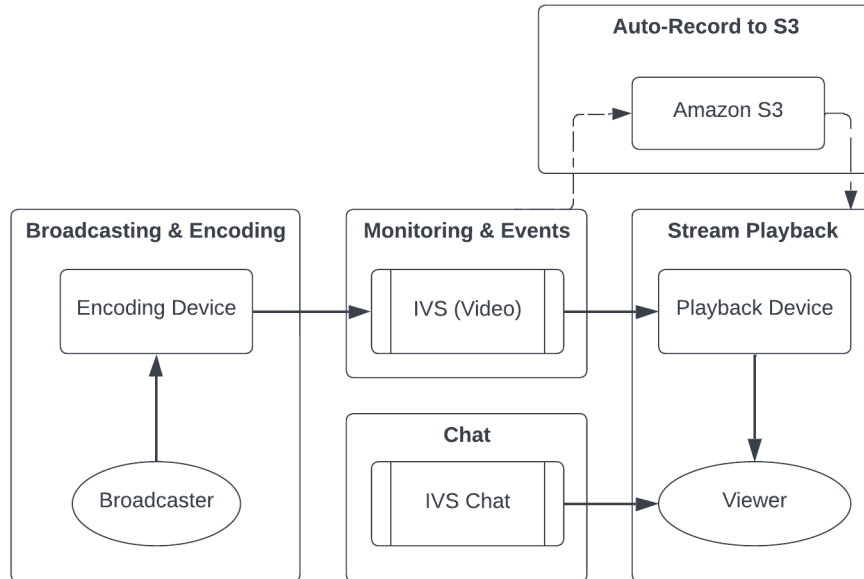
## 测试视频流

请务必验证您的视频流是否正常工作。

在 [Amazon IVS 控制台](#) 中导航到视频流，观察正在流传输的内容并管理实时流。

## 问题排查常见问题

本文档说明了 Amazon Interactive Video Service ( IVS ) 的最佳实践和问题排查提示。使用 IVS 时可能会出现非预期或意外行为。这些行为可能发生在串流过程的不同时间点 ( 包括从广播到内容回放 ) ：



有关支持和其他 Amazon IVS 资源的信息，请参阅 [资源与支持](#)。

## 广播和编码

这一部分的问题涉及广播、编码和流式传输到 IVS 的第一英里 ( 先决 ) 条件。这些行为发生在内容到达 IVS 服务器之前。

主题：

- [the section called “什么是流匮乏？”](#)
- [the section called “为什么流突然停止了？”](#)
- [the section called “当我在流式传输时切换网络后会发生什么？”](#)
- [the section called “如何通过 IVS 实现多区域冗余？”](#)
- [the section called “如何对 IVS Web 广播 SDK 会话进行问题排查？”](#)
- [the section called “如何使用 Google Chrome 的 WebRTC-internals 指标来评估 IVS Web 广播 SDK 会话？”](#)

## 什么是流匮乏？

“流匮乏”是指在您将内容发送到 IVS 时（即 IVS 摄取内容）时，内容包传输出现延迟或中断。如果 IVS 在摄取内容时没有获得编码设备公布的将在特定时间范围内发送的预期比特量，则构成匮乏事件。通常，匮乏事件是由广播者的编码器、本地网络状况和/或在编码设备和 IVS 之间通过公共互联网传输时引起的。

从观众的角度来看，匮乏事件可能以视频延迟、缓冲或冻结的形式出现。流匮乏事件的持续时间可能较为短暂（小于 5 秒），也可能较长（几分钟），具体取决于匮乏事件的性质。

为了允许监控饥饿事件，IVS 将饥饿事件作为亚马逊 EventBridge 事件发送；参见 Amazon 与 Amazon IVS 搭配 EventBridge 使用中的[示例：直播健康状况变化](#)。这些事件将在流进入或退出匮乏状态时发送。您可以根据具体应用场景采取适当的措施，例如将间歇性串流问题通知广播者和观众。

有关其他饥饿监控工具，请参阅监控 [Amazon IVS 低延迟流媒体](#)、IVS [ListStreams](#) API 终端节点（按运行状况筛选）和 IVS [GetStream](#) 终端节点（用于分析单个流）。另请参阅 [the section called “如何监控流匮乏事件？”](#)

## 为什么流突然停止了？

以下是流突然停止（即流会话结束）的最常见原因：

- 缺失摄取数据 – 当摄取流会话完全停止（未将任何数据摄取到 IVS）30 秒时，IVS 摄取服务器将终止 IVS 流会话。30 秒的时间便于广播者重新连接到摄取服务器。但在某些情况下（例如切换网络时），可能无法重新连接到现有的流会话，因为 RTMPS 的 TLS 握手已中断。常见的根本原因包括网络问题（例如广播设备和 IVS 之间的堵塞）、广播设备上的互联网完全中断或广播设备未生成内容片段（FLV 标签）。

流匮乏事件通常会涉及流断开；传入数据停止时会触发匮乏事件。如果发送了一条匮乏开始事件，然后又发送了一条流结束事件（但没有匮乏结束事件），则通常表明该流由于没有向 IVS 发送任何数据而结束。

- IVS StopStream 端点 — 在 IVS 直播会话期间，如果进行了 [StopStream](#) API 调用，IVS 直播会话将结束。StopStream 端点断开传入的 RTMPS 流与 IVS 采集服务器的连接。根据所使用的编码软件/硬件，可能会尝试新的流会话。
- 编码器错误 – 编码过程中出现错误时，某些软件/硬件编码器会断开流会话。从 IVS 的角度来看，这似乎是广播者故意断开连接。但在编码日志中，可能会确定流是因意外错误而断开的。



## 当我在流式传输时切换网络后会发生什么？

当广播公司切换网络（例如，从蜂窝网络切换 WiFi 到蜂窝网络）时，正在进行的 RTMPS 连接就会断开。虽然广播者的互联网连接可能在 3-4 秒后重新建立，但由于网络切换会导致新连接具有新的 IP 地址，从而生成新的 RTMPS 连接。在此切换期间，之前的 RTMPS 连接并未完全断开：编码器不会向 IVS 发送连接断开消息。结果，IVS 会等待 30 秒钟才重新建立之前的 RTMPS 连接，这会阻止新网络上的新 RTMPS 流连接到 IVS。

为了在网络之间实现更快的切换，我们建议您在设备切换网络时使用 IVS [StopStream](#) 端点关闭之前的直播会话。在这种情况下，当广播设备连接到新网络时，广播设备可以调用 StopStream 端点来结束现在处于休眠状态的直播。成功 StopStream 呼叫后，广播设备无需等待 30 秒即可在新网络上开始新的直播会话。

## 如何通过 IVS 实现多区域冗余？

IVS 内的冗余可以通过多种方式实现；请参阅 IVS 安全性中的 [故障恢复能力](#)。

IVS 分为不同的网络面板，即控制面板和数据面板。

- 控制面板是区域性的（基于亚马逊云科技区域），存储有关 IVS 资源（通道、流密钥、回放密钥对和录制配置）的信息。
- 数据面板不受亚马逊云科技区域限制，它是从摄入到出口的数据传输网络。即使在 us-west-2 区域创建了一个通道（例如），流式传输到该通道的视频也可能无法通过 us-west-2 传输。

令请参阅 [全球解决方案](#)、[区域控制](#)。考虑以下两种场景：

- 如果仅使用一个控制面板区域（例如 us-east-1）– 如果特定的亚马逊云科技控制区域出现性能下降或故障，IVS 控制面板在创建、读取、更新或删除以下任何项目时可能会出现延迟或错误：通道、流密钥、回放密钥对或录配配置。如果在故障期间尝试启动新的流，则可能会导致启动流会话时出现更多延迟或错误。根据性能降级的严重程度，或许可以继续向已有正在传输的流的通道进行广播。

如果启用了 [回放授权](#)，当前观众也许可以继续回放正在传输的流，但如果回放密钥对授权出现问题，新观众可能无法开始观看。如果未启用回放授权，则当前观众和新观众都应该能够观看正在传输的流。

出现故障时，IVS 自动录制到 S3 功能也可能中断。

如果发生区域性故障，IVS 控制面板不会自动失效转移到其他亚马逊云科技区域。

- 如果使用两个控制面板区域（例如 us-east-1 和 us-west-2），而在主区域不可用时失效转移到第二个区域，由于 IVS 并未原生支持区域控制面板失效转移，一旦控制面板区域出现问题，则启动新的流或调用控制面板时可能会遇到问题。但数据面板可能不会受到影响，因此控制面板区域正在传输的流将继续传输，不会出现问题。将控制面板移至辅助（失效转移）区域需要在应用程序端完成。您可以编写自定义实现逻辑来处理控制面板失效转移。我们没有关于如何管理区域通道失效转移的官方指导。

分离视频数据面板和区域控制面板增强了 IVS 架构的故障恢复能力：如果区域控制面板出现故障，正在传输的流基本上不会中断。IVS 维持 99.9% 的正常运行时间 SLA，并致力于为客户保证基础设施的稳定性（参见我们的 [SLA](#)）。

## 如何对 IVS Web 广播 SDK 会话进行问题排查？

[IVS Web 广播 SDK](#) 的工作原理与普通的 IVS RTMPS 摄取会话略有不同。Web 广播 SDK 利用 WebRTC 协议流式传输到 IVS 端点。内容进入 IVS 端点后，将对其进行处理并重新封装/转码到 HLS 输出中以供查看。

考虑到 Web 广播 SDK 的性质，请注意以下有关对编码行为进行问题排查的提示：

- 关闭广播设备上所有在广播会话期间不需要打开的选项卡/程序。多余的选项卡/程序可能会占用计算资源（例如 CPU、RAM 和网络），从而导致广播应用程序性能不佳。对于无法关闭的选项卡/程序，请确保其不占用不必要的计算资源。
- 确保设备的上传速度超过 200Kbps。（Web 广播 SDK 的[已知问题](#)中指出了这一点。）要评估上传速度，请打开广播设备的任务管理器，分析流式传输时可用的网络。如果上传速度/比特率低于预期值，请评估可能消耗带宽的其他选项卡/进程。另外，请查看本地网络上是否有其他可能消耗大量带宽的计算机。
- 如果 CPU 使用率出现随机峰值，请查看计算机的任务管理器以了解哪些进程可能消耗 CPU。导致 CPU 使用率出现随机峰值的常见服务是反病毒软件，它会在计算机上定期进行扫描。
- 尝试通过 <https://stream.ivs.rocks/> 进行流式传输，以帮助隔离环境并确保应用程序逻辑不会导致不良行为。该站点由 IVS 运营，是一个可靠的测试环境，可用于评估与 Web 广播 SDK 集成的某部分是否是不良行为的根本原因。
- 尝试使用 Google Chrome 的 WebRTC-internals（见下文）。

## 如何使用 Google Chrome 的 WebRTC-internals 指标来评估 IVS Web 广播 SDK 会话？

通过 IVS Web 广播 SDK 进行流式传输时，广播的编码和发送过程中可能会出现各种行为。按照以下步骤对广播设备上的会话进行问题排查或收集有关会话的信息：

1. 在 Google Chrome 中，打开广播网页。
2. 打开一个新的 Chrome 选项卡并前往 `chrome://webrtc-internals/`（全部复制）。
3. 在原始广播网页选项卡中，启动 Web 广播 SDK 会话并运行，直到行为出现。
4. 观察到行为后，切换到 `chrome://webrtc-internals/` 选项卡（不要结束广播会话），并确保网页显示正确：

► Create Dump

Read stats From:

**Note:** computed stats are in []. Experimental stats are marked with an \* at the end and do not show up in the getStats result.

<https://stream.ivs.rocks/> [ rid: 3067, lid: 1, pid: 32946 ]

[GetUserMedia Requests](#)

```
https://stream.ivs.rocks/, { iceServers: [], iceTransportPolicy: all, bundlePolicy: max-bundle, rtcpMuxPolicy: require, iceCandidatePoolSize: 0 }
```

```
ICE connection state: new
Connection state: new
Signaling state: new
ICE Candidate pair: (not connected)
► ICE candidate grid
```

### Stats Tables

Filter statistics by type including

- certificate (id=CF9C:62:D5:A8:03:45:55:A5:00:F7:0A:59:1D:AA:23:46:DE:31:45:AE:A2:48:6A:03:66:FC:2B:81:2F:2B:32:AD)
- data-channel (id=D1)
- track (id=DEPRECATED\_TO1)
- track (id=DEPRECATED\_TO2)
- local-candidate (candidateType=host, tcpType=active, id=l8m+mV7dh)
- local-candidate (candidateType=host, id=I9P+Kok6N)
- local-candidate (candidateType=host, id=IIUCiN2O)
- local-candidate (candidateType=host, tcpType=active, id=ISkSIIgsx)
- local-candidate (candidateType=host, id=IUWMOuTJ)
- local-candidate (candidateType=host, tcpType=active, id=lwKxg6czL)
- outbound-rtp (kind=audio, mid=1, ssrc=1134012001, id=OT01A1134012001)
- outbound-rtp (kind=video, mid=0, ssrc=3966401599, id=OT01V3966401599)
- peer-connection (id=P)
- media-source (kind=audio, id=SA2)
- media-source (kind=video, id=SV1)
- transport (id=T01)
- Stats graphs for track (id=DEPRECATED\_TO1)
- Stats graphs for outbound-rtp (kind=audio, mid=1, ssrc=1134012001, id=OT01A1134012001)
- Stats graphs for outbound-rtp (kind=video, mid=0, ssrc=3966401599, id=OT01V3966401599)
- Stats graphs for peer-connection (id=P)
- Stats graphs for media-source (kind=audio, id=SA2)
- Stats graphs for media-source (kind=video, id=SV1)

5. 打开屏幕顶部的创建转储可扩展部分。
6. 选择屏幕顶部（Create Dump 正下方）的“下载 PeerConnection 更新和统计数据”，从相关会话中下载 .txt 文件。

7. 下载后，该文件将显示 WebRTC 连接的历史视图。您可以在各种工具中查看此视图，也可以将其发送给 AWS Support 团队进行进一步分析。

## 监控和事件

这一部分的问题与 IVS 监控、指标和事件有关。

主题：

- [the section called “如何监控流匮乏事件？”](#)
- [the section called “如何使用 Amazon CloudWatch 来监控 IVS 服务配额？”](#)
- [the section called “如何使用 IVS 流运行状况诊断流不稳定的问题？”](#)

### 如何监控流匮乏事件？

我们建议使用以下方法来监控流匮乏事件：

- [Amaz EventBridge on with Amazon IVS](#) — 当直播饥饿事件开始或结束时，IVS 会生成直播运行状况变化事件 EventBridge。使用 Amazon EventBridge 目标和规则，您可以使用这些直播饥饿事件在发生直播饥饿时获得警报。有关目标和规则的详细信息，请参阅 [Amazon EventBridge 用户指南](#)。
- [监控 Amazon IVS 低延迟直播功能](#) – 在直播流会话期间，系统会记录数据，然后可通过 IVS 流运行状况分析获取。这包括有关编码器配置、摄取指标和流会话事件的信息。这对于监控正在传输的流或流的追溯评估很有用。您可以使用 IVS 控制台或 API 来识别出现过匮乏的流。即使通道已被删除，流会话数据仍会在 60 天内可用，因此这对于识别过去发生匮乏事件的流很有用。
- [按 Health 筛选直播](#) — 使用 IVS 控制台或 IVS [ListStreams](#) API 端点，您可以使用 health 筛选器来查找处于状态的直播会话。STARVING 此外，的 IVS CloudWatch 指标还 ConcurrentStreams 包括一个 Health 维度，您可以使用该维度来收集处于直播短缺状态的直播总数。请参阅 [监控 Amazon IVS 低延迟直播功能](#)。
- 您可以使用 IVS [GetStream](#) 端点来分析单个数据流。

另请参阅 [the section called “什么是流匮乏？”](#)

### 如何使用 Amazon CloudWatch 来监控 IVS 服务配额？

您可以使用 Amazon CloudWatch 主动监控/管理 IVS 服务配额。参阅 [IVS 服务限额](#)。本文档包含有关为使用量指标创建 CloudWatch 警报的信息。

我们建议您设置适当的 SNS 主题，以便在触发警报时通知相关个人/组。如果警报已触发且限额可调，则应使用新值申请上调服务限额。有关申请上调限额的信息，请参阅 [IVS 服务限额](#)。

## 如何使用 IVS 流运行状况诊断流不稳定的问题？

我们建议您使用 IVS 流运行状况控制面板来评估流不稳定的问题。相关操作说明详见 [监控 Amazon IVS 低延迟直播功能](#)。

此控制面板会显示视频比特率、帧率和音频比特率的时间序列图；示例如下。此外，您也可以单击“查看” CloudWatch，在 Amazon 中查看数据 CloudWatch。

下面将讨论几种场景。

### 互联网带宽不足或互联网堵塞

在这种情况下，即使比特率降低，流也相对不稳定。可能是广播者与 ISP 之间或 ISP 与 IVS 之间的带宽不足，或者通往 IVS 的网络路径出现问题。要解决此问题，请检查是否有其他网络进程在使用带宽，或者联系 ISP 进行网络诊断。

IVS 流运行状况控制面板：

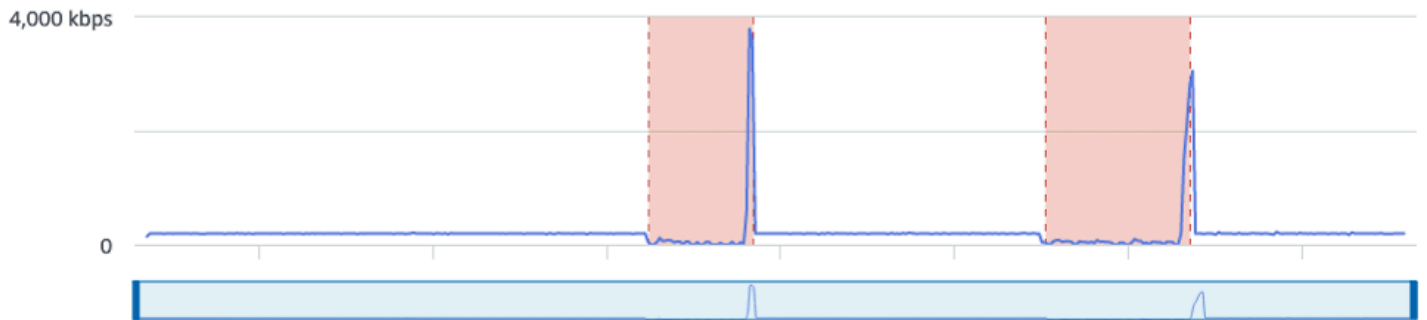
**Video bitrate**



**Frame rate**



**Audio bitrate**



**CloudWatch:**

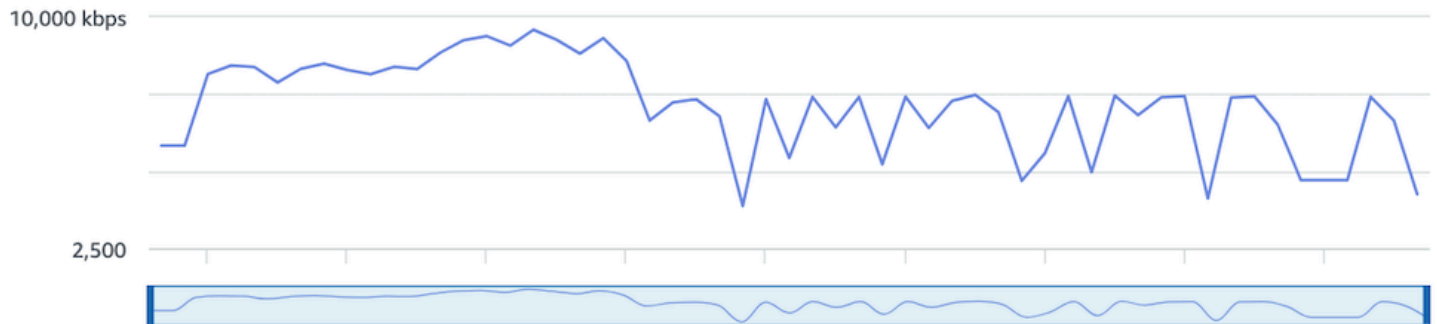


## 比特率过高

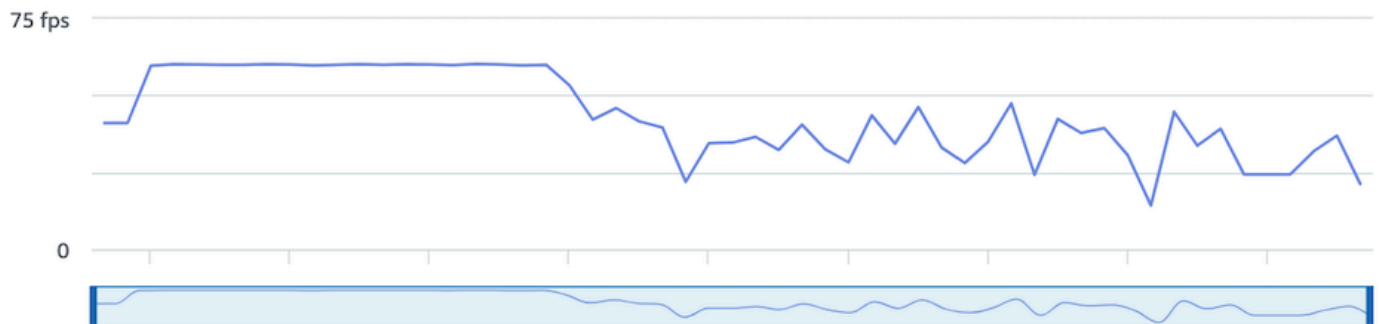
更高的比特率并不意味着更好的质量；在此例中，高比特率导致了不稳定。在许多情况下，由于网络堵塞，高比特率会导致整个广播中的流不稳定。遵循 [the section called “分辨率/比特率/FPS”](#) 中列出的最大比特率。

IVS 流运行状况控制面板：

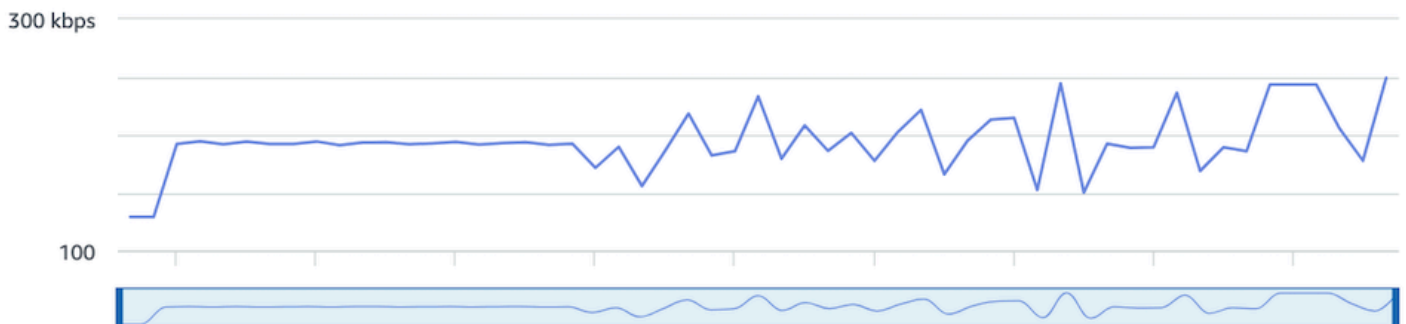
### Video bitrate



### Frame rate



### Audio bitrate



CloudWatch:



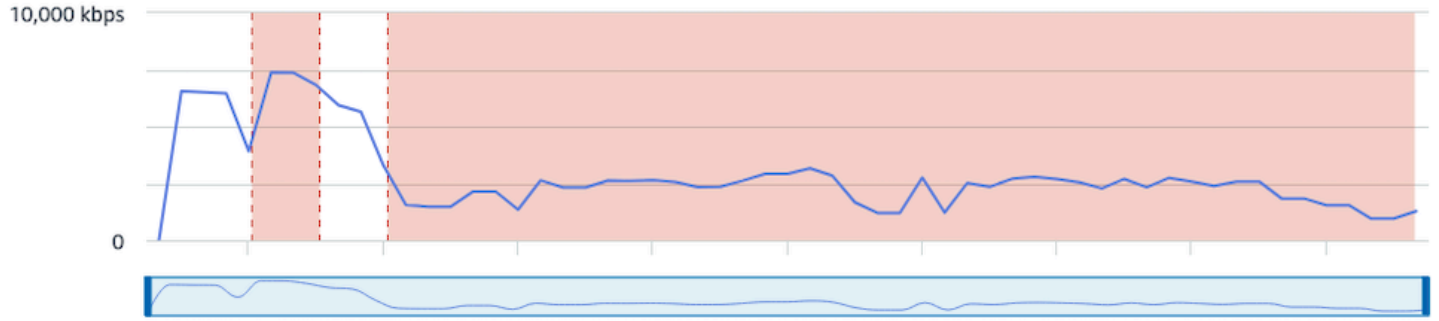
## 网络或硬件问题

视频编码需要大量的计算资源，有时进行视频编码的设备无法承受负载。在这种情况下，请检查设备是否超载（一次运行太多负载）以及编码器是否过时。建议切换到 CPU 利用率较低的编码预设值。

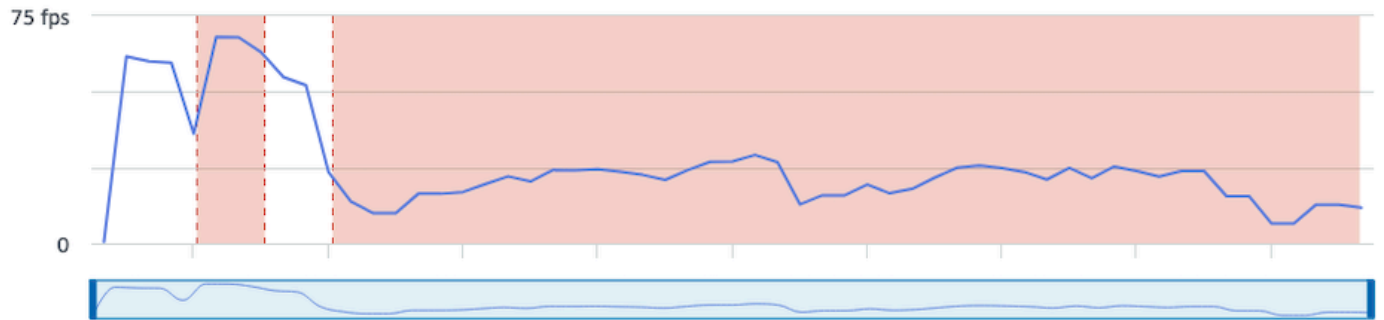
IVS 流运行状况控制面板：



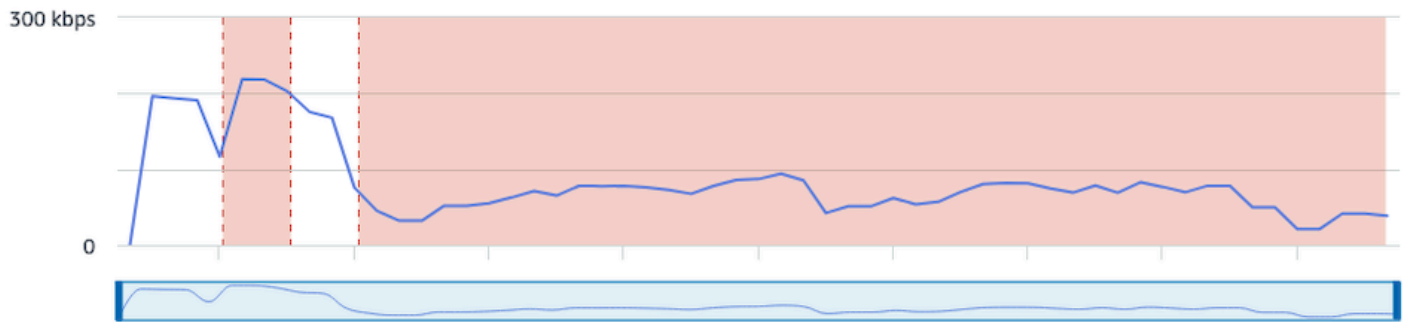
### Video bitrate



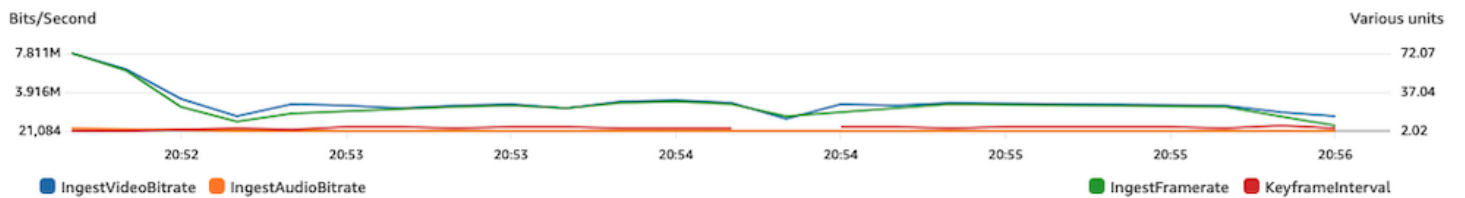
### Frame rate



### Audio bitrate



### CloudWatch:

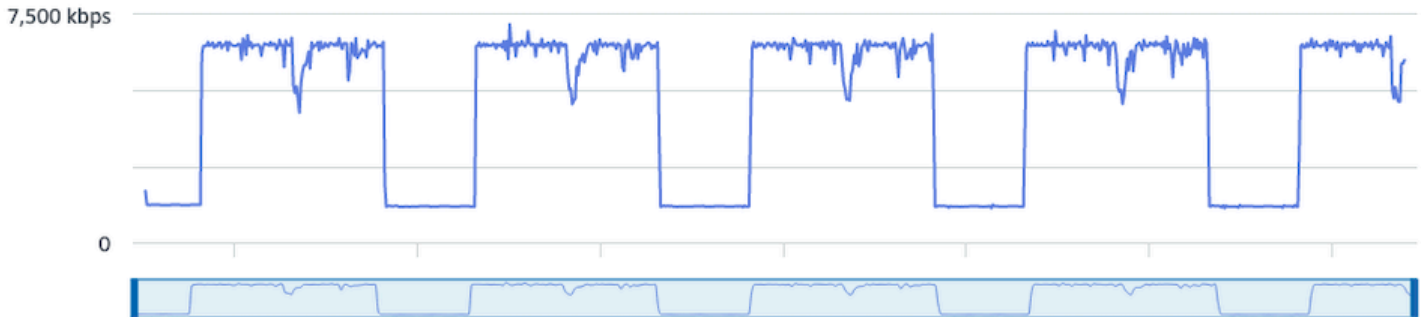


## 比特率峰值和低谷

有时，串流编码器会尝试过度智能化和优化比特率，这通常取决于所压缩的帧的复杂性。如果比特率快速波动，观众可能会因尝试加载过多的数据而遇到缓冲问题。确保启用恒定比特率 ( CBR ) ，因为无论帧复杂程度如何，它都能在整个流中保持稳定的比特率。此外还可能发生低谷的问题；这可能说明您的设备没有足够的 CPU 处理能力让编码器压缩视频。

IVS 流运行状况控制面板：

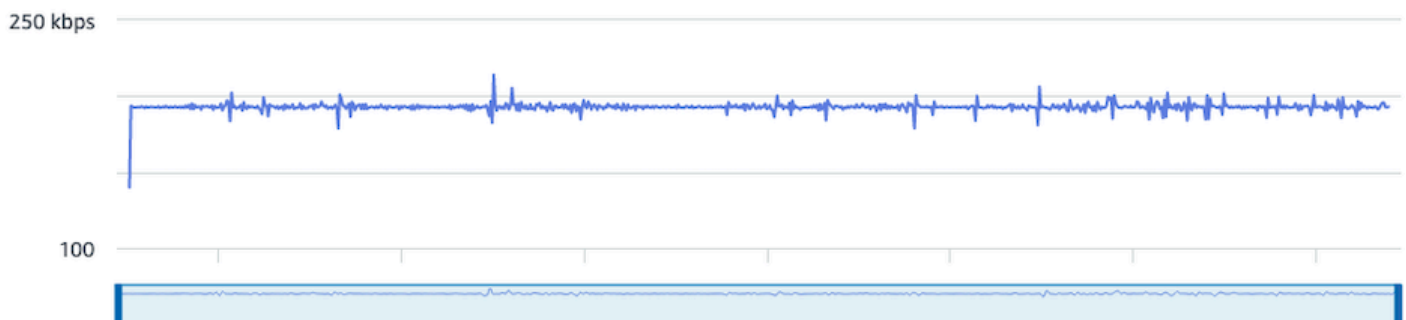
### Video bitrate



### Frame rate



### Audio bitrate



CloudWatch:

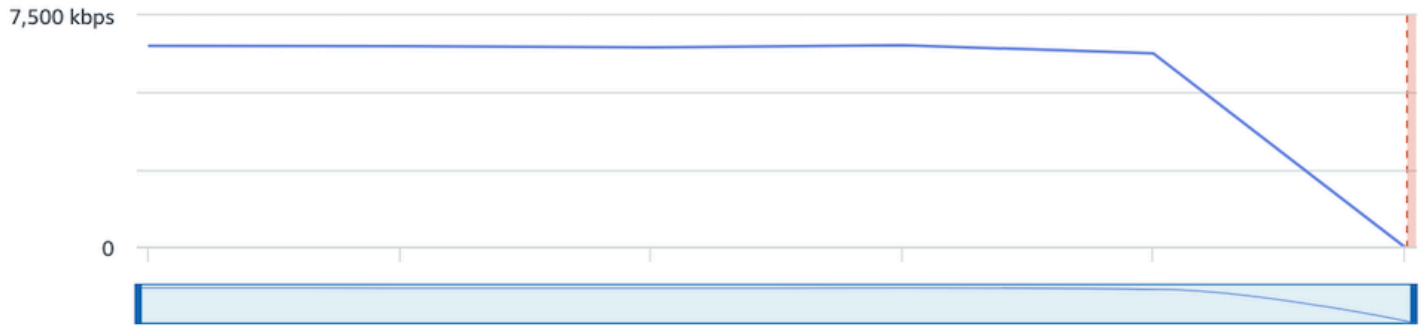


## 互联网连接断开

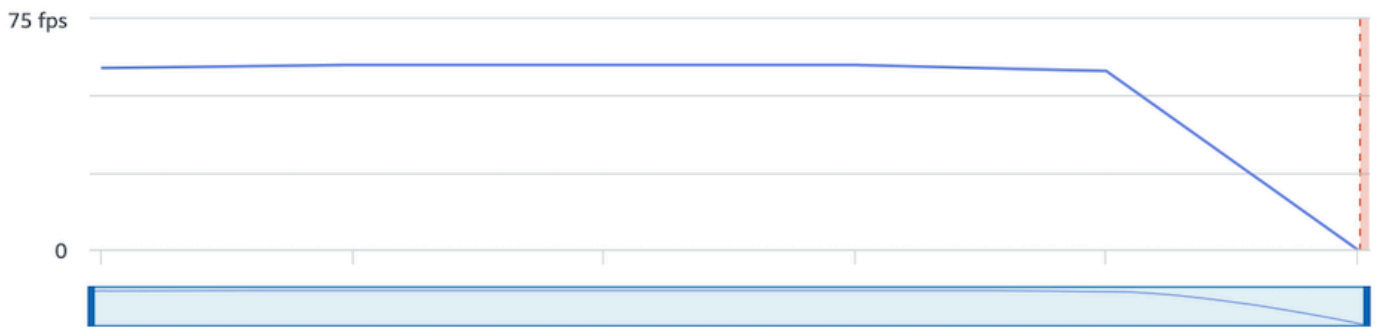
当广播设备遇到互联网问题时，IVS 服务器会用 30 秒的时间来评估是否重新建立了相同的连接。如果未重新建立相同的连接，IVS 服务器将结束流会话。如果互联网连接丢失，一些编码器会尝试重新连接到广播会话，在这种情况下，可能会在初始流结束后启动新的流会话。

IVS 流运行状况控制面板：

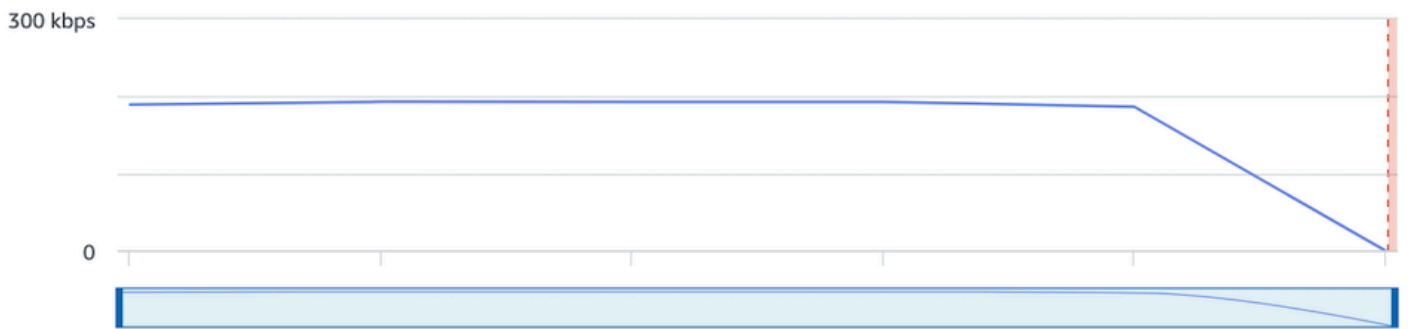
**Video bitrate**



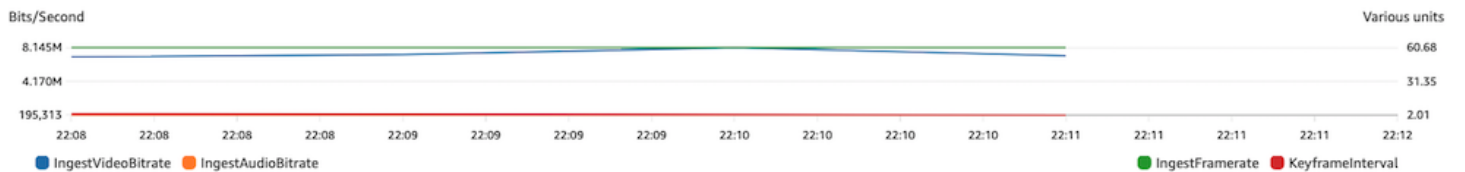
**Frame rate**



**Audio bitrate**



**CloudWatch:**



# 流回放

这一部分的大多数信息都是 IVS 回放器 SDK 特定的，可能不适用于其他回放器。有关更多信息，请参阅 [Amazon IVS 回放器](#)：

主题：

- [the section called “如何调试 IVS 回放器的行为？”](#)
- [the section called “为什么所有观众的回放都冻结/停止？”](#)
- [the section called “是什么导致了 IVS 回放器缓冲？”](#)

## 如何调试 IVS 回放器的行为？

要启用详细日志记录以帮助调试 IVS 回放器，请使用 `setLogLevel` 回放器方法。使用 `DEBUG` 参数修改回放器的日志等级；然后 IVS 回放器将围绕 IVS 回放器上发生的状态和逻辑生成详细的日志记录。

要使用 IVS 回放器进行快速测试，无论是否启用 `DEBUG` 日志，请使用 <https://debug.ivsdemos.com/> 测试站点。如果通过设置菜单启用了 `DEBUG` 日志，则可以在浏览器控制台视图中查看日志。

## 为什么所有观众的回放都冻结/停止？

如果所有观众的回放在内容中同时冻结/停止，这可能是上游行为导致的。通常，根本原因在于广播编码器。

[流匮乏](#)或不良的广播编码器行为可能会同时影响所有观众。如果广播编码断开连接并启动新的流会话，则所有观看者将停止同时接收内容。在评估此行为时，我们建议您根据 [监控 Amazon IVS 低延迟直播功能](#) 中的说明来评估流会话。

## 是什么导致了 IVS 回放器缓冲？

在回放直播流视频和音频时，“缓冲”意味着回放设备无法在需要回放内容之前下载内容。缓冲可以通过多种方式表现出来：内容可能随机停止并开始（也称为卡顿），内容也可能长时间停止（也称为冻结），或者回放器可能进入了 `BUFFERING` 状态。

缓冲的原因有很多，我们可以将其分为三大类：

- 观众端缓冲通常是指单个观众或少量观众受到缓冲事件影响的情形。这些缓冲事件的根本原因通常源于本地网络（LAN）或回放设备问题。如果出现本地网络或设备缓慢的问题，可以通过确保启用自适应比特率回放（ABR）、手动选择较低质量或减少其他程序和设备使用的带宽来解决缓冲问题。

- 网络级缓冲 – 本地网络和 IVS 分发服务器之间（也称为 ISP 级别）可能会出现缓冲问题。在 ISP 级别出现的缓冲行为可能很难进行问题排查，因为可能无法全面了解 ISP。延迟和网络压力（例如，ISP 无法处理整体传入/传出流量）等行为可能会导致向观众提供内容的延迟。
- 广播端缓冲 – 直播流会话的广播端问题可能会导致大规模的观众缓冲问题。例如，当广播设备停止向 IVS 发送数据时，IVS 将没有内容可以传送给回放器，而且 IVS 回放器将在没有下载任何内容时进入缓冲状态。在许多情况下，广播端的缓冲事件会导致大多数（如果不是全部）观众同时受到影响。

## 自动录制到 Amazon S3

有关更多信息，请参阅 [自动录制到 Amazon S3](#)。

主题：

- [the section called “为什么会缺失部分录制内容？”](#)
- [the section called “KMS-S3 加密能否与自动录制到 S3 一起使用？”](#)

### 为什么会缺失部分录制内容？

录制的内容可能因各种原因而缺失。我们建议采取以下步骤排查内容缺失的问题：

1. 确保为所需的 IVS 通道启用了自动录制到 S3：
  - a. 控制台 – 在相关通道的详细信息页面上，检查 General configuration（常规配置）部分，确认自动录制到 S3 的设置是否为 Enabled。如果已启用，则检查 Recording configuration（录制配置），以确认 Storage（存储）和 Recording prefix（录制前缀）的设置均正确。
  - b. CLI – 运行 `get-channel` 并传递所需的 IVS 通道 ARN：

```
aws ivs get-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh"
```

查看是否返回了 `recordingConfigurationArn`。

2. 在指定的 S3 存储桶中查找特定直播会话的录制内容（请参阅 [S3 前缀](#)。）录制会话的 S3 key 前缀位于 Amazon EventBridge [录制状态更改事件](#) 中。注意：如果启用了 [合并流片段](#) 功能，则某些内容可能属于另一个录制的会话。
3. 如果总流时长小于 10 秒或者流内容缺失（即出现流匮乏），则录制的内容可能会缺失，因为没有生成任何内容。

## KMS-S3 加密能否与自动录制到 S3 一起使用？

IVS 自动录制到 Amazon S3 功能不支持 [KMS-S3](#) 加密。尝试使用 KMS-S3 加密时，录制启动将失败并生成“[录制开始失败](#)” EventBridge 事件。推荐的解决方法是使用支持的 [SSE-S3 加密](#)，该加密在上传到 Amazon S3 的所有对象上默认处于启用状态。

## 其他主题

这一部分的问题涉及未纳入其他分类的主题。

主题：

- [the section called ““待验证”错误是什么意思？”](#)
- [the section called “我能否估计使用 IVS 的成本？”](#)

### “待验证”错误是什么意思？

使用 IVS 时，可能会出现以下错误：“Your account is pending verification. 在验证流程完成之前，您可能无法使用此账户执行请求。If you have questions, contact Amazon Web Services Support.”（您的账户正在等待验证。如有疑问，请联系 AWS Support。）

这表明您使用的亚马逊云科技账户必须先通过亚马逊云科技验证，然后才能使用 IVS。（虽然您的账户可以与其他亚马逊云科技服务结合使用，但 IVS 使用增强型验证方法。）

要验证您的 AWS 账户，请通过 AWS Support 中心联系 AWS 账户支持服务，并附上您收到的错误消息：<https://support.console.aws.amazon.com/support/home?#/>

### 我能否估计使用 IVS 的成本？

虽然在流会话之前无法确定使用 IVS 的确切成本，但可以使用下面的大致成本估算器：<https://ivs.rocks/calculator>。更多定价信息详见：<https://aws.amazon.com/ivs/pricing/>。

## 不想要的内容和浏览者

恶意用户可能会试图在您的平台上重新流式传输非预期内容（例如职业体育比赛）。这种流式传输可以显著增加您的应用程序提供的直播视频数量以及相关成本，而不会增加您的业务价值。除了为您提供停止活动流的控制措施外，Amazon IVS 还提供资源，从一开始就帮助检测和防止此类行为。

## 正在检测非预期内容

### 异常检测

您可以检测并提醒流式传输某些非预期内容时出现收视率异常峰值的情况。（一旦检测到峰值，就可以采取[停止流并重置流密钥](#)中提到的步骤，如下所述。）

Amazon CloudWatch 允许您创建警报，这些警报可以在特定情况下发送提醒；例如，当您的收视率激增时。Amazon IVS 会自动向亚马逊 CloudWatch 报告您所有渠道的并发观看次数 (CCV) 指标，因此您只需要设置警报即可。要设置基于 CCV 的异常检测警报，请执行以下步骤：

1. 打开亚马逊 CloudWatch 控制台，[网址为 https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/)。
2. 在左侧导航栏中，依次选择警报和所有警报。
3. 在页面右上角，选择创建警报。
4. 选择选择指标。在“指标”下，选择 IVS，然后选择“全部”，然后选中旁ConcurrentViews边的复选框。
5. 在右下角，选择选择指标。随即打开 4 步警报创建向导。
6. 向导：在步骤 1：指定指标和条件中，指定以下设置：
  - a. 统计数据 = 最大值
  - b. 时间 = 1 分钟
  - c. 阈值类型 = 异常检测
  - d. 每当并发视图为... = 大于该范围
  - e. 异常检测阈值 = 3

此阈值是初始建议值。您可能需要根据典型的流量模式和需求选择不同的值。使用较小值可以更仔细地观察您的指标；使用较大值可减少警报。

- f. 选择下一步。



- 向导：在步骤 2：配置操作中，选择一个现有 SNS 主题或创建一个新 SNS 主题，把电子邮件发送到您指定的地址。要创建发送电子邮件的主题，请选择创建新主题，提供主题名称，输入您的电子邮件地址，然后选择创建主题。选择下一步继续。
- 向导：在步骤 3：添加名称和描述中，为警报添加名称和可选描述，然后选择下一步。
- 向导：在步骤 4：预览和创建中，验证信息是否正确，然后选择创建警报。
- 您的警报已创建。如果出现提示，请按照相关说明确认 SNS 订阅。

有关更多信息，请参阅：

- [监控 Amazon IVS 低延迟直播功能](#)
- [基于异常检测创建 CloudWatch 警报](#)

## 自定义内容审核

您可以探索自定义内容审核解决方案，通过图像识别检测非预期内容。Amazon IVS 能够[自动将 Amazon IVS 实时流录制到 Amazon S3](#)，包括生成用于此类解决方案的缩略图。

考虑以下其他检测和预防技术：

- [使用 Amazon Rekognition 进行 Amazon IVS 审核](#) 的演示展示了如何将 IVS 自动录制到 S3 与 Amazon Rekognition 结合使用来审核实时内容。
- [为您的 Amazon IVS 视频流添加 Hive 内容审核](#)
- [《使用 AI/ML 内容审核创建更安全的在线社区》](#) 是一篇关于在 IVS 应用程序中使用 Amazon Rekognition 的博客文章。

## 防止不想要的内容和浏览者

### 停止流并重置流密钥

如果您检测到某个通道被用来流式传输非预期内容，则可以使用 Amazon IVS 控制台关闭该流：

1. 打开 [Amazon IVS 控制台](#)。（您还可通过 [Amazon 管理控制台](#) 访问 Amazon IVS 控制台。）
2. 如有必要，从导航栏中，使用选择区域下拉菜单选择托管该通道的区域。
3. 选择要停止的流正在运行的通道。
4. 在通道页面上，向下导航到实时流部分，然后选择停止流。

即使您停止流，广播者也可以在该通道上重启流。为防止这种情况，请重置流密钥；这样可以防止广播者在未事先获取新的流密钥的情况下重启流。要重置流密钥，请执行以下操作：

- 还在通道页面上时，向下导航到流配置部分，然后选择重置流密钥。

您也可以通过编程方式停止流并重置（删除/创建）流密钥。请参阅 [Amazon IVS Low-Latency Streaming API Reference](#)。

根据应用程序发放流密钥的方式，您可能需要采取进一步措施来防止获取任何新的流密钥。

## 使用私有通道

在很多情况下，只需将播放 URL 嵌入第三方网站中，就能将非预期内容流式传输给平台外的大量受众。防止这种行为的最佳解决方案是 Amazon IVS 私有通道。通过使用私有通道，您可以将播放受众限制为具有有效播放令牌的观众。播放令牌用于验证播放应用程序中的观众，从而妨碍非预期平台的收视率。此外，您还可以启用源强制功能，防止观众在未托管在您的域上的网站上观看直播。您还可以通过启用严格的源强制功能来扩展此保护以覆盖常见的流式处理应用程序。

请注意，您可以获得私有通道和身份验证的保护，而无需强制用户创建和/或登录正式账户。您的播放应用程序只需在幕后匿名获取令牌即可。您仍然可以利用源强制功能。

要了解有关私有通道的更多信息，请参阅：

- IVS 低延迟流式传输用户指南中的 [设置私有通道](#)。在该文档中，要了解有关源强制功能的更多信息，请参阅 [生成和签名播放令牌](#)。
- [使用 Amazon IVS 为已授权的实时流播放创建私有通道](#)（博客文章）

## 使用播放限制政策

如果您不想使用 [私人频道](#)，您仍然可以通过利用播放限制政策从一些相同的保护措施中受益。这些政策允许您在公共频道上启用诸如 GeoBlocking 和源站强制执行等功能。您可以使用 IVS 控制台或 API 创建播放限制策略，然后将该政策的 ARN 附加到您的频道。

要了解有关播放限制政策的更多信息，请参阅：

- [IVS 低延迟直播入门](#) — 查看有关防止不想要的内容和观看者的信息。
- [IVS 低延迟流媒体 API 参考](#) — 请参阅播放限制策略端点和对象。PlaybackRestrictionPolicy

## 成本（低延迟流式传输）

Amazon IVS 实时视频和自动录制至 S3 功能相关的 Amazon S3 存储具有不同的成本。

### 实时视频

[Amazon IVS 定价](#)模型包含视频输入和输出的不同费用。

视频输入费用取决于您的通道类型。有关通道类型的详细信息，请参阅 IVS 流式传输配置中的[通道类型](#)。

如需帮助为您的使用案例选择正确的通道类型，请使用控制台中的“帮我选择”工具：

1. 在控制台的创建通道页面上，选择自定义配置。
2. 在通道类型下，选择帮我选择。
3. 按照提示进行操作，直到提出建议，然后选择选择建议。

对于视频输出，您需要按小时费率支付传送给观众的视频。费率因分辨率和“计费区域”（视频发送地点）而异。视频输出成本根据使用情况分为几个层次，包括免费套餐。

一个有用的交互式工具是 [IVS 成本估算器](#)。您可以插入通道类型、分辨率、流式传输小时数、查看者数量和计费区域的值。在估算成本时，请注意以下经验法则：

- 查看者变化不定，平均而言，“传送”了 50% 的视频流。成本估算器包括“平均查看者观看持续时间”的选择器。默认值为 50%。预计付费事件的观众人数会更高；即使在这种情况下，并非所有门票持有人都会同时查看。
- 一些观众观看的分辨率低于广播的源分辨率。对于高分辨率流来说尤其如此：一些观众将以较低的分辨率观看，而这没那么贵。这是由于各种查看器限制造成的，包括带宽、网络条件、ISP 和硬件。
- 时间很重要。例如，如果您的流是在学校、工作或假期期间传送，这可能会影响您的受众规模。
- 从非现场用户中构建现场观众是非常困难的。当然，也有例外；引入外部人才（比如有自己的追随者的影响力人物）可以增加受众规模。

### 自动录制到 Amazon S3

对于使用自动录制到 Amazon S3 功能或写入 S3，不收取 Amazon IVS 费用。Amazon S3 存储、Amazon IVS 代表客户进行的 S3 API 调用以及向观众提供存储的视频都需要收费。

## 存储录制视频

客户可以使用 IVS 控制台生成 S3 存储需求和成本的估算。当客户使用控制台来设置通道录制时（创建通道时或稍后），将提供数据使用估算器。这些数据使用估计值可以插入到 [S3 的 AWS 定价计算器](#) 中，以估计 S3 存储和数据移动的每月成本。

在控制台中，创建新通道或编辑现有通道时，选择录制和存储流区域中的启用自动录制。这将显示有关关联成本的信息。

## Record and store streams [Info](#)

### Auto-record to S3 [Info](#)

For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

### Recording configuration

configuration-1 ▼



Create recording configuration

#### State

Active

#### Storage

s3-bucket-name [↗](#)

#### Recording prefix [Info](#)

s3://ivs-r2s3-ivsstoragebucket-1kem14abgbit8/ivs/v1/298083573632/<attached\_channel\_id>/

#### Recorded renditions

All renditions

#### Merge fragmented streams

Disabled

#### Thumbnail recording

At 60-second intervals

#### Thumbnail storage

Store thumbnails sequentially

#### Thumbnail resolution

Source (same resolution as input stream)



#### Associated costs

There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

### ► Tags [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel

Create channel

选择估算数据使用以显示数据使用计算器：

## Estimate data use



### Channel type

The channel type to use in estimations.

Standard

### Average input bitrate

8.5

Mbps

Only use numbers between 0 and 8.5

### Input resolution

1080p

### Input framerate

60fps

### Recording configuration

Choose an existing recording configuration



### Recorded renditions

All renditions

### Thumbnail recording

Record at an interval - 60s

### Target thumbnail resolution

Source (same resolution as input stream)

### Thumbnail storage

Store thumbnails sequentially

### Merge fragmented streams

Disabled

如屏幕中所述，提供的估计值可以与 [AWS 定价计算器](#) 配合使用，以计算 S3 存储和数据移动产生的每月成本的估计值。

## 提供录制视频

向观众提供录制视频的成本取决于所使用的 CDN。如需示例，请参阅 Amazon CloudFront [定价页面](#)。

# 资源和支持 ( 低延迟流式传输 )

## 资源

<https://ivs.rocks/> 是一个专用于浏览已发布内容 ( 演示、代码示例、博客文章 )、估算成本并通过现场演示体验 Amazon IVS 的网站。

[Getting Started with Amazon Interactive Video Service Series' Articles](#) ( Amazon Interactive Video Service 入门系列文章 ) 是关于使用 Amazon IVS 的一系列文章，适用于初学者。这些文章提供了 IVS API 的逐步说明，并在帖子中嵌入了交互式演示。所有演示都可以通过嵌入式 CodePen 直接在帖子中运行。随着时间的推移，这将涵盖各种主题。

Amazon IVS 博客上有很多关于各种主题的文章：

- 在 [AWS Blog](#) 网站上筛选 Amazon IVS，方法是在页面右侧选择 Product or solution ( 产品或解决方案 ) > Media Services ( 媒体服务 ) > Amazon Interactive Video Service。
- 请参阅 DEV 社区网站的[这一部分](#)。

## 演示

有关演示、代码示例和博客文章，请参阅 <https://ivs.rocks/examples>。

## 合作伙伴解决方案

Amazon IVS 与 [Amazon 合作伙伴网络 \( APN \)](#) 中的第三方提供商合作，提供技术解决方案以增强实时流应用程序。有几种类型的合作伙伴解决方案领域：

这种类型的合作伙伴：	提供的解决方案可以做到……
分析	为您的实时流视频应用程序提供运营和业务见解。反过来，这些见解可以提高观众的参与度，并发现提高投资回报率的机会。
交互性	帮助提高实时流视频应用程序受众的参与度。
面部和背景滤镜	使广播者能够更改呈现给实时流受众的面部或背景。



## 分析

[Bitmovin's Analytics](#) 是一项完全托管的服务，具有专为 Amazon IVS 播放器构建的[分析收集器](#)。Analytics 使您能够跟踪并监控跨设备回放运行状况、了解观众的人口统计信息、监控回放体验的质量，并快速发现影响观众的问题。

Bitmovin's Analytics 可通过所有通道收集的可操作数据，根据受众、体验质量 ( QoE ) 和主要错误的指标控制面板帮助提高观众的参与度和留存率。

使用这项服务，您可以访问大约 40 个指标、30 个滤镜和细分。还可以通过 Bitmovin's API 和数据导出获得 200 个维度和滤镜。

要将 Bitmovin's Analytics 与 Amazon IVS 回放器 SDK 集成，请参阅以下入门指南：[Android](#) 和 [iOS](#)。

## 交互性

[LiveLike](#) 提供了一个即用型参与平台，可以在短短几周内提升在线用户体验。通过增加注册、互动、曝光量和赞助，提高每位用户的平均收入。我们的 NASCAR 案例研究显示了一些结果，比如注册人数同比增长 70% ( 2022 年与 2021 年相比 )。使用我们的解决方案，在平台上创建交互式和引人入胜的体验，从而减少观众流失、提高留存率。要将 LiveLike 与 Amazon IVS 集成，请参阅以下博客文章：[LiveLike 快速指南：如何增强实时流交互性](#)。

## 面部和背景滤镜

DeepAR 是一家科技公司，为数字产品团队构建增强现实 ( AR ) 基础设施。从初创公司到上市公司，各种规模的企业都使用我们的软件为全球数十亿用户提供世界一流的 AR 体验。要将 DeepAR 与 Amazon IVS 集成，请参阅 [Amazon IVS 集成](#) 上的 DeepAR 页面。

[BytePlus Effects](#) 结合了庞大的 AR 效果、贴纸和滤镜库，为应用程序开发人员提供了一切可以推动与受众进行更深入互动的工具。要将 BytePlus 与 Amazon IVS 集成，请参阅以下博客文章：[如何使用 BytePlus Effects 和 Amazon IVS 通过实时 AR 效果提高用户参与度](#)。

Camera Kit 是 Snap AR 的 SDK，支持合作伙伴在其应用程序和网站中利用 Snap AR 技术。借助 Camera Kit，企业可以将客户体验提升到一个全新水平，并打造适用于 Snap 底层 AR 技术的全新应用程序。要使用 Snap 的 Camera Kit SDK 将 Snap AR 详解与 Amazon IVS 集成，请参阅以下博客文章：[Unlocking creator expressions to enhance live streaming experiences with Amazon IVS and Snap's Camera Kit AR SDK](#)。

## 支持

[AWS Support Center](#) 提供了一系列计划，通过这些计划可以获取各种工具和专业知识来为您的亚马逊云科技解决方案提供支持。所有支持计划均提供全天候客户服务。要获取可规划、部署和改善您的亚马逊云科技环境的技术支持服务和更多资源，请选择一项最适合您的亚马逊云科技使用案例的支持计划。

[Amazon Premium Support](#) 是一对一的快速响应支持通道，可帮助您在亚马逊云科技中构建和运行应用程序。

[Amazon re:Post](#) 是一个基于社区的问答网站，供开发人员讨论与 Amazon IVS 相关的技术问题。

[联系我们](#) 包含关于您的账单或账户的非技术性查询的链接。如有技术问题，请使用上述开发论坛或支持连接。

## 术语表

另请参阅 [AWS 术语表](#)。在下表中，LL 代表 IVS 低延迟直播；RT 代表 IVS 实时直播。

租期	描述	LL	RT	聊天
AAC	高级音频编码。AAC 是有损数字音频 <a href="#">压缩</a> 的音频编码标准。AAC 旨在成为 MP3 格式的继任者，在相同的比特率下，其音质通常比 MP3 更高。ISO 和 IEC 已将 ACC 标准化，作为 MPEG-2 和 MPEG-4 规范的一部分。	✓	✓	
自适应比特率流	自适应比特率 ( ABR ) 流允许 IVS 播放器在连接质量下降时切换到较低的 <a href="#">比特率</a> ，并在连接质量提高时切换回较高的比特率。	✓		
自适应流	请参阅 <a href="#">通过联播分层编码</a> 。		✓	
管理用户	对 AWS 账户中可用的资源和服务具有管理权限的 AWS 用户。请参阅《AWS 设置用户指南》中的 <a href="#">术语</a> 。	✓	✓	✓
ARN	<a href="#">Amazon 资源名称</a> ，这是 AWS 资源的唯一标识符。具体的 ARN 格式取决于资源类型。有关 IVS 资源使用的 ARN 格式，请参阅《服务授权参考》。	✓	✓	✓
纵横比	描述帧宽度与帧高的比率。例如，16:9 是与全高清或 1080p <a href="#">分辨率</a> 相对应的宽高比。	✓	✓	
音频模式	针对不同类型的移动设备用户及其使用的设备进行优化的预设或自定义音频配置。请参阅 <a href="#">IVS 广播 SDK：移动音频模式（实时直播）</a> 。		✓	
AVC、H.264、MPEG-4 第 10 部分	高级视频编码，也称为 H.264 或 MPEG-4 第 10 部分，是有损数字视频 <a href="#">压缩</a> 的视频压缩标准。	✓	✓	

租期	描述	LL	RT	聊天
背景替换	一种 <a href="#">相机滤镜</a> ，让实时流创作者能够更改其背景。请参阅 IVS 广播 SDK：第三方相机滤镜（实时直播）中的 <a href="#">背景替换</a> 。		✓	
比特率	每秒传输或接收的比特数的直播指标。	✓	✓	
广播，广播者	<a href="#">流</a> 、 <a href="#">直播工具</a> 的其他术语。	✓		
缓冲	播放设备在需要播放内容之前无法下载该内容时出现的一种情况。缓冲可以通过多种方式表现出来：内容可能随机停止并开始（也称为卡顿），内容也可能长时间停止（也称为冻结），或者 IVS 播放器可能进入了暂停状态。	✓	✓	
字节范围播放列表	比标准 <a href="#">HLS 播放列表</a> 更精细的播放列表。标准 HLS 播放列表由 10 秒的媒体文件组成。对于字节范围播放列表，片段持续时间与为 <a href="#">流</a> 配置的 <a href="#">关键帧间隔</a> 相同。  字节范围播放列表仅适用于自动录制到 <a href="#">S3 存储桶</a> 的广播。其是在 <a href="#">HLS 播放列表</a> 之外创建的。请参阅自动录制到 Amazon S3（低延迟直播）中的 <a href="#">字节范围播放列表</a> 。	✓		
CBR	恒定比特率，编码器的一种速率控制方法，无论广播期间发生什么情况，都能在视频的整个播放过程中保持一致的比特率。可以填充操作中的间歇以达到所需的比特率，并且可以通过调整编码质量以匹配目标比特率来量化峰值。我们强烈建议使用 CBR 而不是 <a href="#">VBR</a> 。	✓	✓	
CDN	内容分发网络，一种地理分布式解决方案，通过让直播视频等内容更靠近用户所在位置来优化内容的交付。	✓		

租期	描述	LL	RT	聊天
频道	存储直播配置的 IVS 资源，包括 <a href="#">摄取服务器</a> 、 <a href="#">流密钥</a> 、 <a href="#">播放 URL</a> 和录制选项。流传输工具使用与通道关联的流密钥来启动广播。广播期间生成的所有指标和 <a href="#">事件</a> 都与通道资源相关联。	✓		
通道类型	确定 <a href="#">通道</a> 允许的 <a href="#">分辨率</a> 和 <a href="#">帧速率</a> 。请参阅 IVS Low-Latency Streaming API Reference 中的 <a href="#">Channel Types</a> (通道类型)。	✓		
聊天记录	一个高级选项，可以通过将日志记录配置与某个 <a href="#">聊天室</a> 相关联来启用。			✓
聊天室	一种 IVS 资源，用于存储聊天会话的配置，包括 <a href="#">消息审核处理程序</a> 和 <a href="#">聊天记录</a> 等可选功能。请参阅 Getting Started with IVS Chat 中的 <a href="#">Step 2: Create a Chat Room</a> 。			✓
客户端合成	使用 <a href="#">主机</a> 设备混合舞台参与者的音频和视频流，然后将其作为复合流发送到 IVS <a href="#">通道</a> 。这样可以更好地控制 <a href="#">合成</a> 的外观，代价是客户端资源的利用率更高， <a href="#">舞台</a> 或 <a href="#">主机</a> 问题影响查看者的风险也更高。  另请参阅 <a href="#">服务器端合成</a> 。	✓	✓	
CloudFront	Amazon 提供的 <a href="#">CDN</a> 服务。	✓		
CloudTrail	一项 AWS 服务，用于收集、监控、分析和保留来自 AWS 和外部来源的事件和账户活动。请参阅使用 <a href="#">AWS 记录 IVS CloudTrail API 调用</a> 。	✓	✓	✓
CloudWatch	一项 AWS 服务，用于监控应用程序、响应性能变化、优化资源使用并提供有关运行状况的见解。您可以使用 CloudWatch 监控 IVS 指标；请参阅 <a href="#">监控 IVS 实时流和监控 IVS 低延迟流媒体</a> 。	✓	✓	✓
合成	将来自多个来源的音频和视频流合并为单个流的过程。	✓	✓	

租期	描述	LL	RT	聊天
合成管道	合并多个流并对生成的流进行编码所需的一系列处理步骤。	✓	✓	
压缩	使用比原始表示形式更少的比特数对信息进行编码。任何特定的压缩都是无损或有损的。无损压缩通过识别和消除统计冗余来减少比特数。无损压缩不会丢失任何信息。有损压缩通过删除不必要或不太重要的信息来减少比特数。	✓	✓	
控制层面	存储有关 IVS 资源 ( 例如 <a href="#">通道</a> 、 <a href="#">舞台</a> 或 <a href="#">聊天室</a> ) 的信息，并提供用于创建和管理这些资源的界面。其具有区域性 ( 基于 AWS <a href="#">区域</a> ) 。	✓	✓	✓
CORS	跨源资源共享 ( CORS ) ，这是一项 AWS 功能，允许在一个域中加载的客户端 Web 应用程序与另一个域中的资源 ( 例如 <a href="#">S3 存储桶</a> ) 进行交互。可以根据标头、HTTP 方法和源域配置访问权限。请参阅《Amazon Simple Storage Service 用户指南》中的 <a href="#">使用跨源资源共享 ( CORS ) –Amazon Simple Storage Service</a> 。	✓		
自定义图像源	IVS 广播 <a href="#">SDK</a> 提供的界面，允许应用程序提供自己的图像输入，而不仅限于预设相机。	✓	✓	
数据层面	将数据从 <a href="#">摄取</a> 传输到出口的基础设施。其根据 <a href="#">控制面板</a> 中管理的配置运行，不限于 AWS 区域。	✓	✓	✓
编码器、编码	将视频和音频内容转换为适合直播的数字格式的过程。编码可以基于硬件，也可以基于软件。	✓	✓	
事件	IVS 向 AmazonEventBridge 监控服务发布的自动通知。事件代表直播资源 ( 例如 <a href="#">舞台</a> 或 <a href="#">合成管道</a> ) 的状态或运行状况变化。参见 <a href="#">使用 Amazon EventBridge 进行 IVS 低延迟流媒体</a> 和 <a href="#">使用 Amazon 进行 IVS 实时流 EventBridge 式传输</a> 。	✓	✓	✓

租期	描述	LL	RT	聊天
FFmpeg	一个免费的开源软件项目，其中包含一套用于处理视频和音频文件以及流的库和程序。 <a href="#">FFmpeg</a> 提供了一种跨平台解决方案来录制、转换以及直播音频和视频。	✓		
片段化的流	当广播断开连接，然后在 <a href="#">通道</a> 录制配置中指定的间隔内重新连接时创建。生成的多个流被视为单个广播，并合并到录制流中。请参阅自动录制到 Amazon S3 (低延迟直播) 中 <a href="#">合并片段化的流</a> 。	✓		
帧率	每秒传输或接收的视频帧数的直播指标。	✓	✓	
HLS	HTTP 实时流 (HLS)，一种基于 HTTP 的 <a href="#">自适应比特率流</a> 通信协议，用于向查看者传送 IVS 流。	✓		
HLS 播放列表	组成流的媒体片段列表。标准 HLS 播放列表由 10 秒的媒体文件组成。HLS 还支持更精细的 <a href="#">字节范围播放列表</a> 。	✓		
Host	向舞台发送视频和/或音频的实时事件 <a href="#">参与者</a> 。		✓	
IAM	Identity and Access Management，这是一项 AWS 服务，允许用户安全地管理身份和访问 AWS 服务和资源，包括 IVS。	✓	✓	✓
提取	用于从主机或广播者接收视频流以进行处理或传送给查看者或其他参与者的 IVS 过程。	✓	✓	
提取服务器	接收视频流并将其传送到转码系统，在该系统中，视频流被 <a href="#">转码多路复用</a> 或 <a href="#">转码</a> 为 <a href="#">HLS</a> 以传送给查看者。  摄取服务器是特定的 IVS 组件，用于接收 <a href="#">通道</a> 流以及摄取协议 ( <a href="#">RTMP</a> 、 <a href="#">RTMPS</a> )。有关创建通道的信息，请参阅 <a href="#">IVS 低延迟直播入门</a> 。		✓	

租期	描述	LL	RT	聊天
隔行视频	仅传输和显示后续帧的奇数行或偶数行，从而在不消耗额外带宽的情况下实现 <a href="#">帧速率</a> 的翻倍。出于视频质量方面的考虑，我们不建议使用隔行视频。	✓	✓	
JSON	JavaScript 对象表示法，一种开放标准的文件格式，它使用人类可读的文本来传输由属性值对和数组数据类型或其他可序列化值组成的数据对象。	✓	✓	✓
关键帧、增量帧、关键帧间隔	关键帧（也称为帧内编码或 i 帧）是视频中图像的全帧。后续帧，即增量帧（也称为预测帧或 p 帧），仅包含已更改的信息。关键帧将在一个 <a href="#">流</a> 中多次出现，具体取决于编码器中定义的关键帧间隔。	✓	✓	
Lambda	一项 AWS 服务，用于运行代码（称为 Lambda 函数），无需预置任何服务器基础设施。Lambda 函数可以响应事件和调用请求运行，也可以根据计划运行。例如，IVS 聊天功能使用 Lambda 函数来启用 <a href="#">聊天室的消息审核</a> 。	✓	✓	✓
延迟、glass-to-glass延迟	数据传输中的延迟。IVS 将延迟范围定义如下： <ul style="list-style-type: none"> <li>低延迟：低于 3 秒</li> <li>实时延迟：低于 300 毫秒</li> </ul> <p>G lass-to-glass 延迟是指从摄像机捕捉直播到直播出现在观众屏幕上的延迟。</p>	✓	✓	
通过联播分层编码	支持同时编码并发布具有不同质量级别的多个视频流。请参阅实时直播优化中的 <a href="#">自适应直播：通过联播分层编码</a> 。		✓	
消息审核处理程序	允许 IVS 聊天功能客户能够在用户聊天消息传送到 <a href="#">聊天室</a> 之前自动查看/筛选这些消息。将 <a href="#">Lambda</a> 函数与聊天室关联来将其启用。请参阅 Chat Message Review Handler 中的 <a href="#">Creating a Lambda Function</a> 。			✓



租期	描述	LL	RT	聊天
混合器	IVS 移动广播 <a href="#">SDK</a> 的一项功能，可接收多个音频和视频源并生成单个输出。其支持对代表源的屏幕视频和音频元素进行管理，例如相机、麦克风、屏幕截图以及应用程序生成的音频和视频。然后可以将输出直播到 IVS。请参阅 <a href="#">IVS 广播 SDK：混合器指南（低延迟直播）</a> 中的 <a href="#">配置广播会话以进行混合</a> 。	✓		
多主机直播	将来自多个 <a href="#">主机</a> 的流合并为一个流。可通过使用 <a href="#">客户端</a> 或 <a href="#">服务器端合成</a> 来实现。  多主机直播支持诸如邀请查看者到舞台进行问答、主机之间的竞赛、视频聊天以及主机当众对话等场景。		✓	
多变体播放列表	可用于广播的所有 <a href="#">变体流</a> 的索引。	✓		
OAC	Origin Access Control，一种用于限制对 <a href="#">S3 存储桶</a> 的访问的机制，因此只能通过 <a href="#">CloudFrontCDN</a> 提供诸如录制流之类的内容。	✓		
OBS	Open Broadcaster Software ( OBS )，一款用于视频录制和实时直播的免费开源软件。 <a href="#">OBS</a> 为桌面发布提供了另一种选择 ( IVS 广播 <a href="#">SDK</a> )。熟悉 OBS 的更精密的流传输工具可能会更喜欢这种选择，因为其具有高级制作功能，例如场景过渡、音频混音和图形叠加。	✓	✓	
参与者	以 <a href="#">主机</a> 或 <a href="#">查看者</a> 身份连接到舞台的实时用户。		✓	
参与者令牌	在实时事件 <a href="#">参与者</a> 加入 <a href="#">舞台</a> 时对其进行身份验证。参与者令牌还能控制参与者是否可以向舞台发送视频。		✓	

租期	描述	LL	RT	聊天
播放令牌、播放密钥对	<p>一种授权机制，允许客户限制在<a href="#">私有通道</a>上播放视频。播放令牌由播放密钥对生成。</p> <p>播放密钥对是用于签名和验证查看者播放授权令牌的公有-私有密钥对。请参阅设置私有通道中的<a href="#">创建或导入播放密钥</a>以及 <a href="#">IVS Low-Latency API Reference</a> 中的播放密钥对端点。</p>	✓		
播放 URL	<p>标识查看者用于开始特定<a href="#">通道</a>播放的地址。此地址可以在全球范围使用。IVS 会自动为每个查看者选择 IVS 全球<a href="#">内容分发网络</a>上的最佳位置，以便向每个<a href="#">查看者</a>传送视频。有关创建通道的信息，请参阅 <a href="#">IVS 低延迟直播入门</a>。</p>	✓		
私有通道	<p>允许客户使用基于<a href="#">播放令牌</a>的授权机制来限制对其流的访问。请参阅设置私有通道中的<a href="#">私有通道的工作流</a>。</p>	✓		
逐行视频	<p>按顺序传输并显示每帧的所有行。我们建议在广播的所有阶段使用渐进式视频。</p>	✓	✓	
配额	<p>AWS 账户使用的 IVS 服务资源或操作的最大数量。也就是说，除非另有说明，否则这些限制针对每个 AWS 账户。所有限额都是按区域强制执行的。请参阅 AWS General Reference Guide 中的 <a href="#">Amazon Interactive Video Service endpoints and quotas</a>。</p>	✓	✓	✓

租期	描述	LL	RT	聊天
区域	<p>提供对实际位于特定地理区域的 AWS 服务的访问权限。区域提供容错能力、稳定性和弹性，还可以减少延迟。使用区域，您能够创建保持可用且不受区域中断影响的冗余资源。</p> <p>大多数 AWS 服务请求都与特定的地理区域相关联。除非您明确使用 AWS 服务提供的复制功能，否则在一个区域中创建的资源在任何其他区域中都不存在。例如，Amazon S3 支持跨区域复制。某些服务（例如 <a href="#">IAM</a>）没有跨区域资源。</p>	✓	✓	✓
解决方案	描述单个视频帧中的像素数，例如，全高清或 1080p 定义了具有 1920x1080 像素的帧。	✓	✓	
根用户	AWS 账户的所有者。根用户对 AWS 账户中的所有 AWS 服务和资源具有完全访问权限。	✓	✓	✓
RTMP、RTMPS	实时消息协议，一种通过网络传输音频、视频和数据的行业标准。RTMPS 是 RTMP 的安全版本，通过传输层安全性协议（TLS/SSL）连接运行。	✓	✓	
S3 存储桶	存储在 Amazon S3 中对象的集合。许多策略（包括访问和复制）都是在存储桶级别定义的，适用于存储桶中的所有对象。例如，IVS 广播作为多个对象存储在 S3 存储桶中。	✓		
SDK	软件开发工具包，供开发人员使用 IVS 构建应用程序的库的集合。	✓	✓	✓
自拍分割	允许使用特定于客户端的解决方案替换实时流中的背景，该解决方案接受相机图像作为输入，并返回一个掩码，该掩码为图像的每个像素提供置信度分数，指示图像是在前景还是背景中。请参阅 IVS 广播 SDK：第三方相机滤镜（实时直播）中的 <a href="#">背景替换</a> 。		✓	

租期	描述	LL	RT	聊天
Semantic version	Major.Minor.Patch 形式的版本格式。不影响 API 的错误修复会增加补丁版本，向后兼容的 API 添加/更改会增加次要版本，不向后兼容的 API 更改会增加主要版本。	✓	✓	✓
服务器端合成	<p>使用 IVS 服务器混合舞台参与者的音频和视频，然后将此混合视频发送到 IVS <a href="#">通道</a>，以服务于更多观众或将其存储在 <a href="#">S3 存储桶</a> 中。服务器端合成减少了客户端负载，提高了广播的弹性，并可以更有效地使用带宽。</p> <p>另请参阅<a href="#">客户端合成</a>。</p>		✓	
服务限额	一项 AWS 服务，可帮助您从一个位置管理多个 AWS 服务的 <a href="#">限额</a> 。除了查找配额值，您也可以从 Service Quotas 控制台请求提高配额。	✓	✓	✓
服务相关角色	与 AWS 服务直接关联的一种独特类型的 <a href="#">IAM</a> 角色。服务相关角色由 IVS 自动创建，并包含该服务代表您调用其他 AWS 服务所需的一切权限，例如访问 <a href="#">S3 存储桶</a> 。请参阅 IVS 安全性中的 <a href="#">对 IVS 使用服务相关角色</a> 。	✓		
舞台	IVS 资源，代表实时事件参与者可以在其中实时交换视频的虚拟空间。请参阅 IVS 实时直播入门中的 <a href="#">创建舞台</a> 。		✓	
舞台会话	第一个参与者加入 <a href="#">舞台</a> 时舞台会话开始，最后一个参与者停止向舞台发布几分钟后舞台会话结束。长期存在的舞台在其生命周期内可能有多个会话。		✓	
流	表示从源持续发送到目的地的视频或音频内容的数据。	✓	✓	
流密钥	创建 <a href="#">通道</a> 时由 IVS 分配的标识符；用于授权直播到该通道。将流密钥视为秘密，因为任何拥有它的人都可以直播到通道。请参阅 <a href="#">IVS 低延迟直播入门</a> 。	✓		

租期	描述	LL	RT	聊天
流匮乏	<p>向 IVS 的直播延迟或停止。当 IVS 未收到编码设备宣传的其将在特定时间范围内发送的预期比特量时，就会发生这种情况。出现流匮乏会导致流匮乏事件。</p> <p>从查看者的角度来看，流匮乏可能表现为视频延迟、缓冲或冻结。流匮乏的持续时间可能较为短暂（小于 5 秒），也可能较长（几分钟），取决于导致流匮乏的具体情况。请参阅问题排查常见问题中的<a href="#">什么是流匮乏</a>。</p>	✓	✓	
流传输工具	向 IVS 发送视频或音频流的个人或设备。	✓	✓	
订阅者	接收主机视频和/或音频的实时事件参与者。请参阅 <a href="#">什么是 IVS 实时直播</a> 。		✓	
标签	分配给 AWS 资源的元数据标签。标签可帮助您标识和组织 AWS 资源。在 <a href="#">IVS 文档登录页面</a> 上，请参阅任何 IVS API 文档中的“标记”（适用于实时直播、低延迟直播或聊天）。	✓	✓	✓
第三方相机滤镜	可以与 IVS 广播 SDK 集成的软件组件，允许应用程序在将图像作为 <a href="#">自定义图像源</a> 提供给广播 SDK 之前对其进行处理。第三方相机滤镜可以处理来自相机的图像、应用滤镜效果等。	✓	✓	
缩略图	从流中拍摄的缩小尺寸的图像。默认情况下，每 60 秒生成一次缩略图，但可以配置更短的时间间隔。缩略图分辨率取决于 <a href="#">通道类型</a> 。请参阅自动录制到 Amazon S3（低延迟直播）中的 <a href="#">录制内容</a> 。	✓		

租期	描述	LL	RT	聊天
定时元数据	<p>与流中特定时间戳相关联的元数据。可以使用 IVS API 以编程方式进行添加，并与特定帧相关联。这样可以确保所有查看者在与视频流相关的同一时间点上接收元数据。</p> <p>定时元数据可用于触发客户端上的操作，例如在体育赛事期间更新球队统计数据。请参阅<a href="#">将元数据嵌入视频流中</a>。</p>	✓		
转码	将视频和音频从一种格式转换为另一种格式。传入流可以多个比特率和分辨率将其转码为不同格式，以支持一系列播放设备和网络条件。	✓	✓	
转码多路复用	将 <a href="#">摄取</a> 的流简单地重新打包到 IVS，无需重新编码视频流。“Transmux”是转码多路复用的缩写，这是一个改变音频和/或视频文件格式的过程，同时保留部分或全部原始流。转码多路复用转换为不同的容器格式，而不会更改文件内容。与 <a href="#">转码</a> 不同。	✓	✓	
变体流	<p>多个不同的质量级别的同一广播的一组编码。每个变体流都编码为单独的<a href="#">HLS 播放列表</a>。可用变体流的索引称为<a href="#">多变体播放列表</a>。</p> <p>IVS 播放器从 IVS 接收到多变体播放列表后，可以在播放期间在变体流之间进行选择，随着网络条件的变化，可以无缝地来回转换。</p>	✓		
VBR	可变比特率，编码器的一种速率控制方法，使用动态比特率，根据所需的详情级别在整个播放过程中发生变化。出于视频质量方面的考虑，我们强烈建议不要使用 VBR；改用 <a href="#">CBR</a> 。	✓	✓	

租期	描述	LL	RT	聊天
查看	<p>会主动下载或播放视频的独特观看会话。观看次数是并发观看次数<a href="#">限额</a>的基础。</p> <p>视图会在查看会话开始播放视频时开始。视图会在查看会话停止视频播放时结束。播放是收视率的唯一指标；不考虑音频级别、浏览器选项卡焦点和视频质量等互动启发式算法。在计算观看次数时，IVS不会考虑个别查看者的合法性，也不会尝试重复计算本地化的收视率，例如单台计算机上的多个视频播放器。请参阅服务限额（低延迟直播）中的<a href="#">其他限额</a>。</p>	✓		
查看者	从接收 IVS <a href="#">流</a> 的人。	✓		
WebRTC	<p>Web 实时通信，一个为 Web 浏览器和移动应用程序提供实时通信的开源项目。。它允许直接通信，无需安装插件或下载本机应用程序，从而允许在网页内部进行音频和视频 peer-to-peer 通信。</p> <p><a href="#">WebRTC</a> 背后的技术是作为开放网络标准实现的，可以在所有主流浏览器中作为 JavaScript 常规API使用，也可以作为原生客户端（例如Android和iOS）的库使用。</p>	✓	✓	

租期	描述	LL	RT	聊天
鞭子	<p><a href="#">WebRTC-HTTP 摄取协议</a>，一种基于 HTTP 的协议，允许将基于 WebRTC 的内容摄取到流媒体服务和/或 CDN 中。<a href="#">WHIP</a> 是一份 IETF 草案，旨在对 WebRTC 摄取进行标准化。</p> <p>WHIP 支持与 <a href="#">OBS</a> 等软件兼容，为桌面发布提供了替代方案 (IVS 广播 <a href="#">SDK</a>)。熟悉 OBS 的更老练的主播可能会更喜欢它，因为它具有高级制作功能，例如场景过渡、音频混音和叠加图形</p> <p>在使用 IVS 广播 SDK 不可行或不可取的情况下，WHIP 也很有用。例如，在涉及硬件编码器的设置中，可能无法选择 IVS 广播 SDK。但是，如果编码器支持 WHIP，您仍然可以直接从编码器发布到 IVS。</p> <p>参见 <a href="#">OBS 和 WHIP Support</a>。</p>		✓	
WSS	<p>WebSocket 安全，一种通过加密 TLS 连接建立 WebSockets 的协议。用于连接到 IVS 聊天端点。请参阅 <a href="#">Getting Started with IVS Chat</a> 中的 <a href="#">Step 4: Send and Receive Your First Message</a>。</p>			✓



# 文档历史记录 ( 低延迟流式传输 )

## Low-Latency Streaming User Guide 变更

变更	说明	日期
<a href="#">服务限额</a>	在“API 调用速率配额”表中，我们添加了StartViewerSessionRevocation 和BatchStartViewerSessionRevocation 。( 这些不是新的端点，但表中缺少这些端点。) 它们与端点位于表格中的同一部分； playback-key-pair 端点类型为“私有频道”。	2024年2月5日
<a href="#">广播 SDK : 安卓 1.14.1、iOS 1.14.1、Web 1.8.0</a>	在 low-latency-streaming 广播 SDK 指南中更新了新版本的版本号和工件链接： <a href="#">Android</a> 、 <a href="#">iOS</a> 和 <a href="#">Web</a> 。在 <a href="#">Amazon IVS 文档登录页面上</a> ，更新了广播 SDK 参考链接以指向新版本。另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。  在 Android 指南中，我们添加了一个新的已知问题 ( 视频大小小于 176x176 )。	2024年2月1日
<a href="#">无令牌播放限制</a>	此版本允许在播放授权之外进行源站执法和地理围栏。更改了几个低延迟流媒体文档：	2024 年 1 月 31 日

- [入门](#)-更新了“第 4 步：创建频道”和“第 8 步：阻止不想要的内容和观看者”。
- [Service Quotas](#)-为新终端节点添加了 TPS 限制，并在“其他配额”中添加了新的配额。
- [不想要的内容和观看者](#)-添加了“使用播放限制策略”。
- [私人频道](#)-更新了控制台导航窗格上播放键的位置。

另请参阅 [API 变更](#)。

### [仅限音频播放](#)

在[播放器概述](#)中添加了[纯音频播放](#)。 2024 年 1 月 25 日

### [Player SDK 1.24.0](#)

已在播放器 SDK 指南中更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和[JW 播放器集成](#)。在[Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。另请参阅此发行版的 Amazon IVS [发布说明](#)。 2024 年 1 月 18 日

在网络指南中，我们添加了一个新部分“仅限音频播放”，并删除了关于不支持该audio-only 格式副本的“已知问题”。

### [对自动录制到 Amazon S3 进行故障排除](#)

在“故障排除”中，我们添加了[“KMS-S3 加密能否与自动录制到 S3 一起使用？”](#) 2024 年 1 月 4 日

## [广播 SDK : 安卓 1.13.4、iOS 1.13.4、Web 1.7.0](#)

在 low-latency-streaming 广播 SDK 指南中更新了新版本的版本号和工件链接：[Android](#)、[iOS](#) 和 [Web](#)。在 [Amazon IVS 文档登录页](#)面上，更新了广播 SDK 参考链接以指向新版本。另请参阅此发行版的 Amazon IVS [发布说明](#)。

2024 年 1 月 3 日

## [拆分聊天 UG](#)

此发行版附带主要的文档更改。我们将聊天信息从《IVS Low-Latency Streaming User Guide》移至新的《IVS Chat User Guide》，该指南位于 [IVS 文档登录页面](#)的现有 IVS 聊天功能部分。

2023 年 12 月 28 日

有关其他文档更改，请参阅 [Document History \( Chat \)](#)。

## [IVS 术语表](#)

扩展了术语表，涵盖 IVS 实时、低延迟和聊天术语。

2023 年 12 月 20 日

## [IAM 托管式策略](#)

添加了两个托管策略，即 IVS ReadOnlyAccess 和 IV FullAccess S。请参阅：

2023 年 12 月 5 日

- 安全性页面上关于 [Amazon IVS 的托管式策略](#) 的新增部分。
- 对 IVS 低延迟直播入门中 [步骤 3：设置 IAM 权限](#) 的更改。

### [广播 SDK : Android 1.13.2、iOS 1.13.2](#)

在 low-latency-streaming 广播 SDK 指南：[Android](#) 和 [iOS](#) 中，更新了新版本的版本号和工件链接。

2023 年 12 月 4 日

在 [Amazon IVS 文档登录页](#) 面上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

### [广播 SDK : Android 1.13.1](#)

在 low-latency-streaming 广播 SDK 指南：[Android](#) 中，更新了新版本的版本号和工件链接。

2023 年 11 月 21 日

在 [Amazon IVS 文档登录页](#) 面上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

### [广播 SDK : Android 1.13.0、iOS 1.13.0](#)

在 low-latency-streaming 广播 SDK 指南：[Android](#) 和 [iOS](#) 中，更新了新版本的版本号和工件链接。

2023 年 11 月 17 日

在 [Amazon IVS 文档登录页](#) 面上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">自动记录至 S3</a>	在“ <a href="#">合并碎片流</a> > 资格”中，我们增加了一个要点：“每个流必须在上一个流完成 10 秒或更长时间后开始。”	2023 年 11 月 17 日
<a href="#">服务器端合成和实时合成录制</a>	<p>在在 <a href="#">IVS 流中启用多台主机</a> 中，我们增加了“广播舞台：客户端与服务器端合成”，并更新了“4. 广播舞台。”</p> <p>在<a href="#">安全</a>中，我们向“基于身份的策略示例 &gt; 使用 Amazon IVS 控制台”中的策略添加了 S3 端点。</p> <p>如需了解其他更改，请参阅<a href="#">文档历史记录 ( 实时直播功能 )</a>。</p>	2023 年 11 月 16 日
<a href="#">播放器 SDK 1.23.0</a>	<p>已在播放器 SDK 指南中更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js 集成</a>和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 11 月 14 日
<a href="#">IVS 播放器和广播 SDK</a>	在 <a href="#">播放器概述</a> 和 <a href="#">广播 SDK 概述</a> 中，我们更新了“平台要求” > “原生平台”，以明确支持哪些 SDK 版本。	2023 年 11 月 9 日
<a href="#">IVS 低延迟直播功能入门</a>	我们更新了 <a href="#">设置 IAM 权限</a> 中的操作过程。	2023 年 10 月 20 日

## [广播 SDK : Web 1.6.0](#)

在 low-latency-streaming 广播 SDK 指南：[Web](#) 中，更新了新版本的版本号和工件链接。

2023 年 10 月 16 日

[Amazon IVS 文档登录页面](#)指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

在《网络指南》的“MediaStream 从设备检索”中，我们还删除了这两max行；最佳做法是仅指定ideal。

## [监控 IVS 低延迟直播功能](#)

重命名了“监控 Live Stream Health”页面，并添加了 CloudWatch “使用监控 IVS”（已作为单独页面删除）中的信息。更新了 CloudWatch 控制台说明。

2023 年 10 月 12 日

## [广播 SDK : Android 1.12.1](#)

在 low-latency-streaming 广播 SDK 指南：[Android](#) 中，更新了新版本的版本号和工件链接。此外还增加了新章节 [使用蓝牙耳机](#)。

2023 年 10 月 12 日

在 [Amazon IVS 文档登录页面](#)上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">播放器 SDK 1.22.0</a>	<p>已在播放器 SDK 指南中更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js 集成</a>和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 10 月 3 日
<a href="#">控制台内流式传输</a>	<p>在低延迟流式传输入门中，我们在<a href="#">步骤 5：设置流式传输软件</a>中添加控制台内流式传输。</p>	2023 年 10 月 2 日
<a href="#">广播 SDK：混合器指南</a>	<p>添加<a href="#">镜像广播</a>并附有 Android 和 iOS 示例。</p>	2023 年 9 月 18 日
<a href="#">广播 SDK：Web 1.5.2</a>	<p>在 low-latency-streaming 广播 SDK 指南：<a href="#">Web</a> 中，更新了新版本的版本号和工件链接。</p> <p><a href="#">Amazon IVS 文档登录页面</a>指向最新版本的广播 SDK 参考。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 9 月 14 日
<a href="#">非预期内容</a>	<p>将故障排除常见问题中的现有内容拆分为自己的顶层页面。</p> <p>在 <a href="#">IVS 低延迟流式传输入门</a> 中，添加了“步骤 8：防止出现非预期内容（推荐）”。</p>	2023 年 9 月 8 日

<a href="#">自动录制到 Amazon S3</a>	在 <a href="#">字节范围播放列表</a> 中，阐明了片段持续时间与为流配置的关键帧间隔相同（不是约 2 秒的固定时长）。	2023 年 8 月 25 日
<a href="#">广播 SDK : Web 1.5.1、Android 1.12.0 和 iOS 1.12.0</a>	<p>在 low-latency-streaming 广播 SDK 指南中更新了新版本的版本号和工件链接：<a href="#">Web</a>、<a href="#">Android</a> 和 <a href="#">iOS</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a> 面上，更新了广播 SDK 参考链接以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 8 月 23 日
<a href="#">播放器 SDK 1.21.0</a>	<p>已在播放器 SDK 指南中更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js</a> 集成和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a> 面，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 8 月 22 日
<a href="#">通道类型定义</a>	更新了通道类型定义以提供更多详细信息，尤其是有关渲染的转码阶梯。请参阅 IVS 流式传输配置中的 <a href="#">通道类型</a> 。	2023 年 8 月 18 日



## [实时流式传输发布](#)

此发行版附带主要的文档更改。我们将之前的文档重命名为 IVS Low-Latency Streaming，并发布了新的 IVS Real-Time Streaming 文档。[IVS 文档登录页面](#) 现在有单独的实时流式传输和低延迟流式传输部分。每个部分都有各自的用户指南和 API 参考。

2023 年 8 月 7 日

我们将一些信息从 IVS Low-Latency User Guide 移至新的 IVS Real-Time User Guide：

- 有关舞台和多台主机的详细信息。
- “Monitoring Stage Health”现已更名为 [Monitoring Real-Time Streaming](#)。

有关其他文档更改，请参阅：

- [Stage API Reference Changes](#)
- [Document History \(Real-Time Streaming\)](#)

## [广播 SDK : Web 1.5.0、Android 1.11.0 和 iOS 1.11.0](#)

已在广播 SDK 指南中更新了新版本的版本号和构件链接：[Web](#)、[Android](#) 和 [iOS](#)。

2023 年 8 月 7 日

在 [Amazon IVS 文档登录页面](#) 上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">设置私有频道</a>	在 <a href="#">Token Schema</a> 中，添加了有关 exp 字段的澄清信息。	2023 年 7 月 31 日
<a href="#">Security: getting IVS status information</a>	在 <a href="#">Incident Response</a> 中，更新了有关获取 IVS 状态的信息，以指向 AWS Health Dashboard。	2023 年 7 月 31 日
<a href="#">Auto-Record to Amazon S3: OAC and CORS</a>	在 <a href="#">Playback of Recorded Content from Private Buckets</a> 中，将来源访问身份 ( OAI ) 替换为源访问控制 ( OAC )。还添加了有关为 CORS 配置 S3 存储桶以回放录制的流的信息。	2023 年 7 月 31 日
<a href="#">资源与支持</a>	在“Partner Solutions”>“Face and Background Filters”中，添加了关于 Camera Kit 的段落。	2023 年 7 月 21 日
<a href="#">Broadcast SDK: Android Guide</a>	次要更改。在简介中，提到不支持仿真器。在“Create the Player and Set Up Event Listener”中将 PlayerActivity class 更改为 Activity。更改了“Thread Safety”中的文本。	2023 年 7 月 21 日

## [R2S3 rendition filtering and thumbnail enhancements](#)

2023 年 7 月 17 日

现在，IVS 客户可以控制录制到 Amazon S3 时所生成的流渲染，以及生成的缩略图分辨率。在 IVS User Guide 中，请参阅：

- [Getting Started with IVS](#)：在“Step 4: Create a Channel”>“Console Instructions”中，更新了屏幕截图和说明。
- [Auto-Record to Amazon S3](#)：在“JSON Metadata Files”中，添加了 latest\_thumbnail 并更新了 thumbnail。在“Thumbnails”和“Discovering the Renditions of a Recording”中，添加了渲染分辨率的描述。
- [Costs](#)：在“Storing Recorded Video”中更新了屏幕截图。

另请参阅 [IVS API 参考更改](#)。

## [播放器 SDK 1.20.0](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2023 年 7 月 14 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [IVS 入门](#)

在[如何禁用录制](#)中，修复了 CLI 示例。

2023 年 7 月 14 日

## [广播 SDK : Web 1.4.0、Android 1.10.0 和 iOS 1.10.0](#)

已在广播 SDK 指南中更新了新版本的版本号和构件链接：[Web](#)、[Android](#) 和 [iOS](#)。

2023 年 7 月 13 日

在 [Amazon IVS 文档登录页面](#)上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [私有通道的查看者会话撤消](#)

IVS 客户现在能够撤消与身份验证令牌关联的查看者会话，以阻止和停止使用该令牌进行播放。有关更多信息，请参阅[设置私有通道](#)：

2023 年 6 月 28 日

- “令牌架构”- 我们添加 viewer-id 并修改了 viewer-session-version 。
- “撤消查看者会话”- 新部分。

另请参阅 [IVS API 参考更改](#)。

## [安全 TLS 更新](#)

在“基础架构安全”>“[API 调用](#)”中，将 TLS 版本更新为最低版本 1.2，推荐版本为 1.3。

2023 年 6 月 27 日

## [广播 SDK : iOS 1.9.1 和 iOS 1.7.5](#)

更新了广播 SDK 指南中新发行版的版本号和构件链接：[iOS](#)。

2023 年 6 月 27 日

[Amazon IVS 文档登录页面](#)指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [广播 SDK : Web 1.3.3](#)

已在广播 SDK 指南中更新新版本的版本号和构件链接：[Web](#)。

2023 年 6 月 16 日

[Amazon IVS 文档登录页面](#)指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [高级通道类型](#)

2023 年 6 月 2 日

引入了两种新的通道类型，ADVANCED\_SD 和 ADVANCED\_HD。我们更新了几个页面：

- [播放器 SDK 概述](#)：在“减少第三方播放器的延迟”中，注明了高级流不需要降低延迟功能
- [广播 Web SDK 指南](#) — “创建 AmazoniVS Broadcast Client 实例”中的更改。
- [广播 Android SDK 指南](#) — “获取推荐的广播设置”中的更改。
- [广播 iOS SDK 指南](#) — “获取推荐的广播设置”中的更改。
- [服务限额](#) — 在“其他限额”> IVS 中，为新通道类型增加了两行“采集比特率”。
- [流媒体配置](#) — “通道类型”中的更改。
- [成本](#) — 增加了新的通道类型并提到了“帮我选择”工具

## [广播 SDK : Android 1.9.0 和 iOS 1.9.0](#)

在广播 SDK 指南中，更新了新版本的版本号和构件链接：[Android](#) 和 [iOS](#)。

2023 年 6 月 1 日

在 [Amazon IVS 文档登录页](#) 面上，更新了广播 SDK 参考链接以指向新版本。

在 [广播 SDK 概述](#) 中，将支持的 iOS 版本从 11+ 更新到了 12+ (适用于没有舞台功能的 SDK)。

在 [iOS 指南](#) 中，增加了一个新章节，“iOS 如何选择相机分辨率和帧率”。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [自动录制到 Amazon S3](#)

在“示例：recording\_ended.json”中，已将 byte\_range\_playlist 值从 byte-range-multivariant.m3u8 更新为 byte-range-variant.m3u8 。

2023 年 5 月 25 日

## [播放器 SDK 1.19.0](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2023 年 5 月 23 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [广播 SDK : iOS 1.8.1 和 iOS 1.7.4](#)

更新了广播 SDK 指南中新发行版的版本号和构件链接：[iOS](#)。

2023 年 5 月 16 日

[Amazon IVS 文档登录页面](#)指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。



## [Monitoring stage health](#)

添加了 [Monitoring Stage Health](#)，这是介绍 Amazon IVS 新功能的“用户指南”新页面。对于舞台运行状况，我们还：

2023 年 5 月 11 日

- 在 [在 IVS 流中启用多台主机](#) 中添加了运行状况信息。
- 在“通过亚马逊 IVS [使用亚马逊 IVS](#)”中添加 [EventBridge](#) 了两个阶段更新事件。
- 在 [IVS 服务限额](#) 中为新端点添加了调用速率限额。

注意：随着 IVS 实时流式传输于 2023 年 8 月 2 日推出，本文档已更名为“监控 Amazon IVS 实时直播功能”，并移至新的 IVS 实时流式传输用户指南。

## [舞台参与者限制](#)

在 [服务限额](#) 中，删除了“舞台参与者”限制。取而代之的是订阅用户和发布者参与者限制。

2023 年 5 月 2 日

## [广播 SDK : Web 1.3.2](#)

已在广播 SDK 指南中更新新版本的版本号和构件链接：

2023 年 5 月 1 日

[Web](#)。

[Amazon IVS 文档登录页面](#) 指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">RTMP 支持：文档勘误表</a>	更改了 <a href="#">Android 广播 SDK 指南</a> 和 <a href="#">iOS 广播 SDK 指南</a> ，以表明这些 SDK 仅支持 RTMPS 摄取（不支持不安全的 RTMP 摄取）。	2023 年 4 月 29 日
<a href="#">舞台参与者限制</a>	此版本包含以下更改： <ul style="list-style-type: none"><li>• <a href="#">启用多台主机</a>：将舞台参与者的最大数量从 12 更新为 1,000。</li><li>• <a href="#">服务限额</a>：将参与者限制更新为 1,000，并为订阅用户和发布者参与者添加了新的限制。更改某些端点的 TPS。</li></ul>	2023 年 4 月 27 日
<a href="#">IVS 用户指南登录页面</a>	在 <a href="#">IVS 是什么？</a> 主页，添加了“多台主机”和“IVS 聊天功能”部分，并更新了“延迟”部分。	2023 年 4 月 27 日
<a href="#">资源与支持</a>	在“合作伙伴解决方案”>“面部和背景滤镜”中，更新了 DeepAR 链接。	2023 年 4 月 25 日
<a href="#">资源与支持</a>	已添加关于合作伙伴解决方案的部分。	2023 年 4 月 17 日
<a href="#">播放器 SDK：Web 指南</a>	已在“已知问题和解决方法”中添加一个问题（Web 播放器不支持 audio_only 渲染）。	2023 年 4 月 17 日
<a href="#">串流配置</a>	已在 <a href="#">隐藏式字幕</a> 中添加链接，可链接到一篇有关字幕的新博客文章。	2023 年 4 月 14 日

## [广播 Web SDK 指南](#)

已进行多项更新：

2023 年 4 月 10 日

- 在“创建 AmazonIVS 实例”中 BroadcastClient，添加了一条关于确保您的客户端配置与后端渠道类型保持一致的注释。
- 在“隐藏视频”代码示例中，已将 VIDEO\_DEVICE\_NAME 更改为 VIDEO\_DEVICE\_NAME.source。
- 在“启用多台主机”中，已将 ConnectionState 参考更改为 StageConnectionState。
- 在“使用 Broadcast SDK 添加多台主机”和“已知问题”中，将信息同步到此处[GitHub](#)。

## [串流配置](#)

在“[视频设置](#)”中，添加了一个 ColorSpace 子弹。

2023 年 4 月 5 日

## [启用多台主机](#)

在[设置 AWS CLI](#)中，已将舞台命名空间从 ivsrealtime 更改为 ivs-realtime。

2023 年 4 月 5 日

## [播放器 SDK 1.18.0](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2023 年 4 月 4 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [RTMP 支持](#)

已在多份文档中说明现在同时支持 RTMP（不安全提取）和 RTMPS。此外，这还会影响提取端点；请参阅[设置流式传输软件](#)、[Android 广播 SDK 指南](#)和 [iOS 广播 SDK 指南](#)。

2023 年 3 月 30 日

## [设置私有频道](#)

在[生成并签署回放令牌](#)中，向负载中添加了可选字段 `single-use-uuid`，用于生成一次性令牌。

2023 年 3 月 29 日

## [广播 SDK : Web 1.3.1](#)

已在广播 SDK 指南中更新新版本的版本号和构件链接：[Web](#)。

2023 年 3 月 28 日

在 [Amazon IVS 文档登录页面](#)上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">在流上支持多台主机</a>	<p>已添加新页面，即在<a href="#">IVS 流中启用多台主机</a>。已在<a href="#">服务限额</a>中添加了“Amazon IVS 舞台”端点，并对其他限额 &gt; Amazon IVS 添加了舞台限制。</p> <p>另请参阅<a href="#">舞台 API 参考更改</a>。</p>	2023 年 3 月 23 日
<a href="#">广播 SDK : Android 1.8.0、iOS 1.8.0 和 Web 1.3.0</a>	<p>已在广播 SDK 指南中更新了新版本的版本号和构件链接：<a href="#">Android</a>、<a href="#">iOS</a> 和 <a href="#">Web</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a>面上，更新了广播 SDK 参考链接以指向新版本。</p> <p>已在<a href="#">广播 SDK 概述</a>中添加舞台平台要求。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 3 月 23 日
<a href="#">Web 广播 SDK</a>	<p>已在<a href="#">已知问题和解决方法</a>中添加一个问题：Safari 广播的观众有时会在视频推送中看到绿色构件。</p>	2023 年 3 月 17 日
<a href="#">广播 SDK : Android 1.7.3</a>	<p>在广播 SDK 指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a>面上，更新了指向新版本的广播 SDK 参考链接。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2023 年 3 月 2 日

<a href="#">播放器 SDK 1.17.0</a>	在所有播放器指南中，更新了新版本的版本号和构件链接： <a href="#">Web</a> 、 <a href="#">Android</a> 、 <a href="#">iOS</a> 、 <a href="#">Video.js 集成</a> 和 <a href="#">JW 播放器集成</a> 。	2023 年 2 月 28 日
	在 <a href="#">Amazon IVS 文档登录页面</a> ，更新了播放器 SDK 参考链接，以指向新版本。	
	另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。	
<a href="#">服务限额</a>	阐明了所有限额都是按区域强制执行的。	2023 年 2 月 24 日
<a href="#">问题排查常见问题</a>	“使用私人通道”中阐明了使用私人通道以防止不良内容的情况。在“广播和编码”中增加了两个小节，分别介绍对 IVS Web 广播 SDK 会话进行问题排查以及使用 Chrome WebRTC-internals。	2023 年 2 月 17 日
<a href="#">用于自动录制到 S3 的字节范围标签和清单文件</a>	在 <a href="#">自动录制到 Amazon S3</a> 中，更新了“录制内容”，并在 recording_started 和 recording_ended 的 JSON 示例中添加了“字节范围播放列表”和新字段。	2023 年 2 月 16 日
<a href="#">IVS 聊天功能入门</a>	一开始，假设 IVS 聊天功能也可以单独使用，无需视频流。请参阅 Amazon IVS Chat User Guide 中的 <a href="#">Getting Started with IVS Chat</a> 。	2023 年 2 月 9 日

<a href="#">问题排查常见问题</a>	<p>增加了有关不良内容的新章节。</p> <p>2023 年 9 月 8 日更新：此部分已移至<a href="#">非预期内容</a>。</p>	2023 年 2 月 6 日
<a href="#">回放器 SDK 概述</a>	在 <a href="#">浏览器和平台要求</a> 中，添加了一条注释，指出类似浏览器的环境不支持 SDK Video.js 和 Player JW 集成。	2023 年 2 月 6 日
<a href="#">自动录制到 Amazon S3</a>	在合并片段化流的 <a href="#">资格要求</a> 中，将所需的比特率差异从 10% 更改为 50%。	2023 年 2 月 6 日
<a href="#">串流配置</a>	修改了 <a href="#">使用 Amazon IVS 广播 SDK 的流</a> 以包含 Web 广播 SDK ( 不仅是 Android 和 iOS ) 。	2023 年 2 月 2 日
<a href="#">IVS 聊天功能客户端消息收发 SDK : Android 1.1.0</a>	<p>在聊天功能 SDK 指南：<a href="#">Android</a> 中，更新了新版本的版本号和构件链接。</p> <p><a href="#">Amazon IVS 文档登录页面</a>指向最新版本的 SDK 参考。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p> <p>此版本包括详尽的 Chat Kotlin 协同教程，分为两部分：</p> <ul style="list-style-type: none"><li>• <a href="#">第 1 部分：聊天室</a></li><li>• <a href="#">第 2 部分：消息和事件</a></li></ul>	2023 年 1 月 31 日

## [聊天功能 Android SDK 教程](#)

为聊天功能客户端消息收发 SDK 添加了详尽的 Android 教程。该教程分为两部分：

2023 年 1 月 24 日

- [第 1 部分：聊天室](#)
- [第 2 部分：消息和事件](#)

## [服务限额](#)

上调了部分聊天功能限额：

2023 年 1 月 19 日

- CreateChatToken、DeleteMessage、DisconnectUser 和 SendEvent 房间的 TPS
- 其他配额：并发聊天连接；DeleteMessage DisconnectUser、和 SendMessage 请求的速率；每次连接的消息请求速率；以及聊天室

## [私有通道](#)

在 [令牌 Schema](#) 中，将 strict-origin-enforcement 字段添加到令牌负载中。

2023 年 1 月 17 日

## [回放器 SDK 1.16.0](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2023 年 1 月 17 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。



<a href="#">聊天功能 React 和 React Native 最佳实践</a>	<p>增加了此新的聊天功能页面。</p> <p>注意：自 2023 年 12 月 28 日起，<a href="#">本文档</a>已移至新的 IVS Chat User Guide。</p>	2023 年 1 月 13 日
<a href="#">聊天功能 React Native SDK 教程</a>	<p>增加了聊天功能客户端消息收发 SDK 的详细 React Native 教程。该教程分为两部分：</p> <ul style="list-style-type: none"><li>• <a href="#">第 1 部分：聊天室</a></li><li>• <a href="#">第 2 部分：消息和事件</a></li></ul>	2023 年 1 月 10 日
<a href="#">故障排除</a>	<p>增加了新的问题排查常见问题页面，其中说明了相关最佳实践和问题排查提示。</p>	2023 年 1 月 6 日
<a href="#">为 record-to-S3 清单文件增加了时间戳</a>	<p>向自动录制到 S3 功能创建的 S3 清单文件添加了时间戳。请参阅 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 12 月 9 日
<a href="#">回放器 SDK 延迟</a>	<p>添加了<a href="#">减少第三方播放器的延迟</a>部分。</p>	2022 年 12 月 8 日
<a href="#">广播 Web SDK 指南</a>	<p>向此页面添加了内容（以前仅启用 GitHub）。</p>	2022 年 12 月 8 日
<a href="#">广播 SDK : Android 1.7.2</a>	<p>在广播 SDK 指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>上，更新了指向新版本的广播 SDK 参考链接。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 12 月 6 日

<a href="#">IVS 设置</a>	<p>在 <a href="#">IVS 入门</a> 中，更新了创建亚马逊科技账户和设置权限的步骤。添加了“步骤 2：设置根用户和管理用户”。</p> <p>在 <a href="#">安全</a> 中，对 IAM 部分的开头进行了细微更改。</p>	2022 年 12 月 5 日
<a href="#">聊天功能：设置和 iOS SDK 教程</a>	<p>在 <a href="#">IVS Chat 入门</a> 中，更新并重命名为“初始设置”。</p> <p>在《用户指南》中添加了 <a href="#">Chat iOS 教程</a> 页面，指向现有的教程 GitHub。</p>	2022 年 12 月 5 日
<a href="#">自动录制到 S3 的成本</a>	在 <a href="#">自动录制到 Amazon S3</a> 中，阐明了成本。	2022 年 12 月 2 日
<a href="#">聊天 JavaScript SDK 教程</a>	<p>为 Chat 客户端消息收发 SDK 添加了详尽的 JS 教程。该教程分为两部分：</p> <ul style="list-style-type: none"><li>• <a href="#">第 1 部分：聊天室</a></li><li>• <a href="#">第 2 部分：消息和事件</a></li></ul>	2022 年 12 月 2 日
<a href="#">Web 回放器已知问题</a>	在播放器 Web SDK 指南中，我们添加了一个 <a href="#">已知问题和解决方法</a> ：在 iOS 移动浏览器上播放静音实时流时，如果恢复处于非活动状态的播放器选项卡，可能会出现播放器不稳定的情况。	2022 年 11 月 18 日

## [设置私有频道](#)

在“创建或导入播放密钥”中，对内容进行了重新组织并阐明了何时使用私有密钥和公有密钥。在“生成和签名播放令牌”中，阐明了您不必在 [jwt.io](#) 中输入公有密钥。

2022 年 11 月 18 日

## [聊天功能日志记录](#)

该项新功能的初始版本。请参阅用户指南中的以下更改：

2022 年 11 月 17 日

- [聊天记录](#) — 新页面。
- [Chat 入门](#) — 更新了 IAM 权限，添加了设置聊天记录的程序。
- [服务限额](#) — 添加了对新端点和日志记录配置的限制。
- CloudWatch — 添加了日志目标指标。

2023 年 10 月 12 日更新：  
此 CloudWatch 文档已删除，内容已移至 [Monitoring IVS 低延迟流媒体](#)。

2023 年 12 月 28 日更新：  
聊天相关 CloudWatch 内容已移至“[监控亚马逊 IVS 聊天](#)”。

## [聊天客户端消息 SDK : JavaScript 1.0.2](#)

Chat SDK 指南中更新了新版本的版本号和工件链接：[JavaScript](#)。

2022 年 11 月 9 日

[Amazon IVS 文档登录页面](#)指向最新版本的 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">直播通道的拆分视图 ( 用于监控直播流运行状况 )</a>	在 <a href="#">访问流式会话数据</a> 中，添加了访问新拆分视图的控制台说明。这是从“直播通道”页面获取会话运行状况数据的更快方法。	2022 年 11 月 8 日
<a href="#">资源与支持</a>	在 DEV 社区网站上添加了 IVS 博客的链接。	2022 年 11 月 7 日
<a href="#">自动录制到 Amazon S3</a>	在“合并片段化的流”>“资格”中，删除了多余的项目符号“源视频质量必须相同”。	2022 年 11 月 7 日
<a href="#">播放器 SDK 1.14.0</a>	<p>在所有播放器指南中，更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js 集成</a>和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 11 月 1 日
<a href="#">播放器 SDK : Web 指南</a>	更新了 <a href="#">使用内容安全策略</a> ，以反映所有浏览器的较新版本均已更新，以处理新的 CSP 规则。删除了旧章节“Hosting Assets on the Same Origin”（在同一源中托管资产）和“Hosting Assets on a Separate Origin”（在独立源中托管资产）。	2022 年 10 月 27 日
<a href="#">Amazon IVS Chat 入门</a>	更新并说明了步骤 3，以前为“验证并授权聊天客户端”，现在为 <a href="#">创建聊天令牌</a> 。	2022 年 10 月 27 日

<a href="#">播放器 SDK : Web 指南</a>	在“Sample Code” ( 示例代码 ) 中，在 PLAYBACK_URL 两边添加了引号并说明应将其替换为 URL 字符串。	2022 年 10 月 24 日
<a href="#">聊天客户端消息 SDK : JavaScript 指南</a>	添加了新章节 <a href="#">React Native Support</a> ( React Native 支持 )。	2022 年 10 月 24 日
<a href="#">IVS 聊天客户端消息传递 SDK : 1.0.1 JavaScript</a>	此新 SDK 的初始版本。请参阅《IVS 用户指南》中的 <a href="#">Amazon IVS Chat 客户端消息收发 SDK</a> 。  <a href="#">Amazon IVS 文档登录页面</a> 指向最新版本的 SDK 参考。  另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。	2022 年 10 月 18 日
<a href="#">广播 SDK : iOS 1.7.1</a>	更新了广播 SDK 指南中新发行版的版本号和构件链接： <a href="#">iOS</a> 。  在 <a href="#">Amazon IVS 文档登录页面</a> 上，更新了广播 SDK 参考链接以指向新版本。  另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。	2022 年 10 月 6 日
<a href="#">Web 播放器 1.13.0 发布说明</a>	在 Web 播放器 1.13.0 的发布说明中增加了有关“Sawmill Enabled” ( 已启用 Sawmill ) 日志的已知问题。	2022 年 9 月 27 日

## [广播 SDK 1.7.0 发行版](#)

在广播 SDK 指南中，更新了新版本的版本号和构件链接：[Android](#)、[iOS](#)。

2022 年 9 月 22 日

在 [Amazon IVS 文档登录页面](#) 上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [播放器 1.13.0 发行版](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2022 年 9 月 20 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [广播 SDK : iOS 1.5.2](#)

更新了广播 SDK 指南中新发行版的版本号和构件链接：[iOS](#)。

2022 年 9 月 12 日

[Amazon IVS 文档登录页面](#)指向最新版本的广播 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [IVS Chat 客户端消息收发 SDK : Android 1.0.0 和 iOS 1.0.0](#)

这些新 SDK 的初始版本。请参阅《IVS 用户指南》中的 [Amazon IVS Chat 客户端消息收发 SDK](#)。

2022 年 9 月 8 日

[Amazon IVS 文档登录页面](#) 指向最新版本的 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

我们更新了《[Amazon IVS Chat 入门](#)》，其中包含各种演示（包括演示令牌生成的后端服务器应用程序）的链接和用于删除聊天消息的示例代码。

## [使用亚马逊监控亚马逊 IVS CloudWatch](#)

对于一些具有 Channel 维度的 Amazon IVS 指标，我们更正了描述。通道值并非 ARN（如前所述）。它们是通道的 resource-id，即 ARN 的最后一部分。

2022 年 9 月 2 日

2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 [Monitoring IVS 低延迟流媒体](#)。

## [资源与支持](#)

在《Amazon IVS 用户指南》中增加了一个新页面。此页面指向有关 Amazon IVS 的其他信息和支持。

2022 年 9 月 1 日

## [合并片段化的流](#)

该项新功能的初始版本。请参阅以下文档变更：

2022 年 8 月 30 日

- Amazon IVS 入门 - 更新了[第 3 步：使用可选录制创建通道](#)中的控制台和 CLI 说明。
- 自动录制到 S3 - 增加了[合并片段化的流](#)
- EventBridge — 在“[示例：记录状态更改](#)”中添加 recording\_session\_id 了 recording\_session\_stream\_ids 字段。

## [监控实时流运行状况](#)

在 [Filter Streams by Health](#) (按运行状况筛选流) 中，更正了 CLI 示例：将 filter-by name 更改为 filter-by health。

2022 年 8 月 17 日



## [扩展 BASIC 通道](#)

BASIC 通道的最大分辨率和比特率已更改。分辨率最高可达 1080p，480p 的比特率最高可达 1.5 Mbps，480p 到 1080p 之间的分辨率最高可达 3.5 Mbps。请参阅以下文档变更：

2022 年 8 月 16 日

- IVS 入门 - 更新了[初始通道设置](#)中的屏幕截图。
- 流传输配置 - 更新了[通道类型](#)中的定义。
- 成本 - 更新了[实时视频](#)中的通道定义。
- 服务限额 - 在[其他限额](#)中，为 BASIC 通道更新了 IVS 的摄取比特率和摄取分辨率信息。

## [播放器 SDK 1.12.0 发行版 : Web](#)

更新了播放器指南中新发行版的版本号和构件链接：[Web](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2022 年 8 月 9 日

[Amazon IVS 文档登录页面](#)指向最新版本的播放器 SDK 参考。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [广播 SDK : iOS 1.5.1](#)

更新了 7 月 28 日发行版的发布说明：增加了一个修复项目（内存泄漏）。

2022 年 8 月 8 日

<a href="#">自动录制到 Amazon S3</a>	在 <a href="#">JSON 元数据文件</a> 中，增加了 recording_started_at 和 recording_ended_at 的注释 ( 关于使用 duration_ms 来确定录制的持续时间 )。	2022 年 8 月 8 日
<a href="#">Amazon IVS 广播 SDK : Web</a>	( 此处和发行说明中 ) 更新了此版本的 7 月 21 日条目，删除了 1.0.0 版本号，并添加了仅在 future 版本中更新的说明。GitHub	2022 年 8 月 4 日
<a href="#">澄清了控制台指令</a>	说明了只有在导航窗格折叠时，单击汉堡包图标才会打开导航窗格。这种情况存在于三个地方： <ul style="list-style-type: none"><li>• <a href="#">IVS 入门</a> –“第 5 步：查看您的直播流”</li><li>• <a href="#">监控直播流</a> –“访问流会话日期”和“按运行状况筛选流”</li></ul>	2022 年 8 月 3 日
<a href="#">广播 SDK 发行版 : iOS 1.5.1</a>	更新了广播 SDK 指南中新发行版的版本号和构件链接： <a href="#">iOS</a> 。  <a href="#">Amazon IVS 文档登录页面</a> 指向最新版本的广播 SDK 参考。  另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。	2022 年 7 月 28 日

- [Amazon IVS 广播 SDK : Web](#) Web 广播 SDK 首次发布。 2022 年 7 月 21 日  
请参阅 [Amazon IVS 文档登录页面](#) 上“Amazon IVS 广播 SDK”下的文档内容。
- 此外还更新了 Amazon IVS 入门中的 [使用 Amazon IVS 广播 SDK 进行流式传输](#)。
- 重要提示：对于未来的版本，文档将仅在以下地址更新 GitHub：<https://aws.github.io/amazon-ivs-web-broadcast/>（不在此处）。
- [IVS Chat 指标](#) 使用亚马逊监控亚马逊 IVS CloudWatch — 为 IVS Chat 添加了一个指标 (Deliveries)。 2022 年 7 月 15 日
- 2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring [IVS 低延迟流媒体](#)。
- [播放器 SDK 发行版 : iOS 1.8.3](#) 在 [iOS 播放器指南](#) 中，更新了新发行版的版本号和构件链接。 2022 年 7 月 14 日
- [Amazon IVS 文档登录页面](#) 指向最新版本的播放器 SDK 参考。
- [估算数据使用量屏幕截图](#) 在 [Cost](#) (成本) 中，“估算数据使用量”的屏幕截图已更新：不再提供“音频”副本。 2022 年 6 月 30 日

## [播放器 SDK 1.11.0 发行版 : Web](#)

2022 年 6 月 28 日

更新了播放器指南中的新版本的版本号和构件链接：[Web](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

在 [Amazon IVS 文档登录页](#)面上，更新了播放器 SDK Web 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

在 [Player SDK: Web Guide](#) ( 播放器 SDK : Web 指南 ) 中，我们从“已知问题和解决方法”中删除了两个不再适用的项目：

- 使用 Video.js 集成在 iOS 移动浏览器上播放录制的内容时，重新播放按钮无法正常工作。
- 在 Google Pixel 4 或 4a 移动浏览器上播放实时流时，回放可能会意外停止。

## [广播 SDK 1.5.0 发行版](#)

2022 年 6 月 22 日

在广播 SDK 指南中，更新了新版本的版本号和构件链接：[Android](#)、[iOS](#)。

在 [Amazon IVS 文档登录页](#)面上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

<a href="#">串流配置</a>	在 <a href="#">Stream Ingest: Codecs, RTMPS, and Port 443</a> (流摄取：编解码器、RTMPS 和端口 443) 中，澄清术语：您指定 IVS 摄取服务器 (其在路径中包括端口 443)。	2022 年 6 月 20 日
<a href="#">服务限额</a>	对于 IVS Chat 配额，增加了“每个房间的 SendMessage 请求速率”配额，并明确了现有的 SendMessage 请求费率配额适用于您的所有房间。	2022 年 6 月 14 日
<a href="#">提取服务器格式</a>	在 <a href="#">Amazon IVS 入门</a> 中，更新了“Final Channel Creation (最终通道创建)”中的屏幕截图，以显示当前格式的提取服务器 (带端口 443 和路径 /app/)。更新了“使用 OBS Studio 进行流传输”和“使用 FFmpeg 对录制视频进行流传输”中的说明。	2022 年 6 月 14 日
<a href="#">播放器开发工具包 1.10.0 发行版：Web 和 Android</a>	<p>更新了播放器指南中的新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">Video.js 集成</a> 和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 5 月 24 日

<a href="#">服务限额</a>	为和添加了通话费率配额。 GetStreamSession ListStreamSessions ( 这些 IVS 端点是之前在启动流运行状况时添加的。 )	2022 年 5 月 16 日
<a href="#">iOS 播放器指南</a>	在“已知问题和解决方法”中，删除了关于 iOS 10 的项目符号，该项目符号不再受支持： <ul style="list-style-type: none"><li>• iOS 10 设备从后台返回时可能会发生崩溃。</li></ul> 解决办法：进入后台前，将层的 player 属性设置为 nil。	2022 年 5 月 10 日
<a href="#">广播开发工具包：自定义图像源</a>	在示例 iOS 应用程序中为新的 CIFilter 实现添加了一个项目符号。	2022 年 5 月 10 日
<a href="#">Web 播放器指南</a>	在“内容安全策略”中，为来自第三方 CDN 的视频流添加了域 ( *.akamaized.net 和 *.ext.cloudfront.live.hls.ttvnw.net )。	2022 年 4 月 29 日
<a href="#">Video.js 播放器指南</a>	在“Events ( 事件 )”中，从允许的 event 值列表删除了 MetadataEventType ( 不再可用 )。	2022 年 4 月 29 日
<a href="#">安全策略更新</a>	在 <a href="#">基于身份的策略示例</a> 中，更改了控制台策略 ( 添加了 Chat、lambda 和 Amazon CloudWatch ) 及其介绍性文本。	2022 年 4 月 29 日

<a href="#">私有通道</a>	在 <a href="#">生成和签名播放令牌</a> 中，指定了令牌架构 payload (有效负载) 字段中的 exp 时间戳值为 UTC。	2022 年 4 月 29 日
<a href="#">OBS Studio 设置</a>	IVS 入门 – 在 <a href="#">使用 OBS Studio 进行流传输</a> 中，阐明了如何指定服务器和流密钥，并添加了设置视频分辨率、比特率和关键帧间隔的步骤。	2022 年 4 月 29 日
<a href="#">流运行状况更新</a>	<p><a href="#">监控 Amazon IVS Live Stream Health</a> — 在“控制台说明”中指出，直播会话详细信息页面中提供了高分辨率 CloudWatch 指标的图表。在“按生命值筛选直播”中，添加了“Health Dimension”</p> <p>ConcurrentStreams。</p> <p>使用亚马逊监控亚马逊 IVS CloudWatch — ConcurrentStreams 指标中添加了一个新维度 (Health)，用于按渠道运行状况筛选结果。</p> <p>2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring <a href="#">IVS 低延迟流媒体</a>。</p>	2022 年 4 月 28 日

## [Amazon IVS Chat](#)

该项新功能的初始版本。新增和更新的信息可通过 [Amazon IVS 文档登陆页面](#) 访问：

2022 年 4 月 26 日

- [Getting Started with Amazon IVS Chat](#) – 新页面（位于 Amazon IVS Chat User Guide 中）。
- [Chat Message Review Handler](#) – 新页面（位于 Amazon IVS Chat User Guide 中）。
- 使用亚马逊监控亚马逊 IVS CloudWatch -- 添加了新的指标和新的聊天命名空间。

2023 年 10 月 12 日更新：  
此 CloudWatch 文档已删除，内容已移至 [Monitoring IVS 低延迟流媒体](#)。

2023 年 12 月 28 日更新：  
聊天相关 CloudWatch 内容已移至“[监控亚马逊 IVS 聊天](#)”。

- [安全](#) – 在“数据保护”中，添加了聊天项目符号。在“身份和访问管理”中，添加了有关“Amazon IVS Chat 基于资源的策略”的章节。在“基础设施安全”中，添加了有关“Amazon IVS Chat”的章节。
- [服务限额](#) – 在“服务限额增加”中，更新了可调整的限额。将两个部分合并为“其他



限额”。在“API 调用速率配额”、“其他配额”和“Service Quotas 与 CloudWatch 使用量指标集成”中添加了聊天信息。

- 在 [Amazon IVS 文档登陆页面](#)，添加了《Amazon IVS Chat》章节，其中包含两个 API 参考文档。请参阅 [IVS Chat API 文档更改](#)（本页的新章节）。

2023 年 12 月 28 日更新：  
我们将聊天相关信息移至新的《IVS Chat User Guide》。有关其他文档更改，请参阅 [Document History \( Chat \)](#)。

### [iOS 播放器 1.8.2 发行版](#)

在 [iOS 播放器指南](#) 中，更新了新发行版的版本号和构件链接。

2022 年 4 月 22 日

在 [Amazon IVS 文档登录页面](#) 上，更新了 iOS 播放器 SDK 参考链接以指向新版本。

### [手动安装 SDK](#)

在 [广播 SDK : Android](#) 和 [播放器 : Android 指南](#) 的“入门”>“安装库”部分增加了有关手动安装的句子。

2022 年 4 月 19 日

<a href="#">广播 SDK 1.4.0 发行版</a>	<p>在广播 SDK 指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a>、<a href="#">iOS</a>。</p> <p>增加了一个有关<a href="#">广播 SDK：自定义图像源</a>的新页面。</p> <p>在<a href="#">Amazon IVS 文档登录页</a>面上，更新了广播 SDK 参考链接以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 4 月 19 日
<a href="#">iOS 播放器 1.8.1 发行版</a>	<p>在<a href="#">iOS 播放器指南</a>中，更新了新发行版的版本号和构件链接。</p> <p>在<a href="#">Amazon IVS 文档登录页</a>面上，更新了 iOS 播放器 SDK 参考链接以指向新版本。</p>	2022 年 3 月 31 日
<a href="#">Android 播放器的设备支持</a>	<p>在<a href="#">Android 播放器指南</a>中，阐述了支持哪些原生 Android 设备（手机和平板电脑）。在<a href="#">播放器概述</a>中，在“原生平台”部分添加了“支持的设备”表格列。</p>	2022 年 3 月 23 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	<p>修改了“会话已结束”事件并更新了相关描述。此外，还阐述了“已创建会话”和“流结束”事件的描述。</p>	2022 年 3 月 18 日
<a href="#">播放器 Video.js 集成</a>	<p>在“使用脚本标签进行设置”中，步骤 1 为示例添加了一个结尾 <code>&lt;/script&gt;</code>。</p>	2022 年 3 月 4 日

### [广播 SDK 1.3.0 版本](#)

在广播 SDK 指南中，更新了新版本的版本号和构件链接：[Android](#)、[iOS](#)。

2022 年 3 月 3 日

在 [Amazon IVS 文档登录页面](#) 上，更新了广播 SDK 参考链接以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

### [播放器 1.8.0 版本](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2022 年 3 月 1 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

### [将亚马逊 EventBridge 与亚马逊 IVS 配合使用](#)

对于录制结束失败事件，添加了示例失败案例：尝试写入主播放列表失败。

2022 年 2 月 10 日

### [将亚马逊 EventBridge 与亚马逊 IVS 配合使用](#)

对于录制开始事件，添加了注释，说明清单文件和视频段需要一段时间才能写入清单文件和视频。

2022 年 2 月 9 日

<a href="#">广播 SDK : Android 1.2.1 版本</a>	<p>在广播 SDK 指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a> 面上，更新了指向新版本的广播 SDK 参考链接。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2022 年 2 月 3 日
<a href="#">适用于播放器 SDK 的 React Native 包装器</a>	在 <a href="#">Player Android 指南</a> 和 <a href="#">玩家 iOS 指南</a> 中，添加了指向新 React Native 包装器的 GitHub 代码和文档的链接。	2022 年 1 月 27 日
<a href="#">适用于播放器 SDK 的 React Native 包装器</a>	在 <a href="#">Player Android 指南</a> 和 <a href="#">玩家 iOS 指南</a> 中，添加了指向新 React Native 包装器的 GitHub 代码和文档的链接。	2022 年 1 月 27 日
<a href="#">Web 播放器 CSP 更改</a>	在“在独立源中托管资产”部分增加了适用于 Chrome 的信息。	2022 年 1 月 25 日
<a href="#">设置私有频道</a>	在“令牌 Schema”部分，增加了有关在 access-control-allow-origin 令牌有效负载字段支持多个域和通配符域的信息。	2022 年 1 月 24 日
<a href="#">Web 播放器 1.7.0 发布说明</a>	<a href="#">发布说明</a> 更新了有关 setInitialBufferDuration() 的项目符号，说明此版本不支持 iOS 移动浏览器。	2022 年 1 月 21 日

## [播放器 1.7.0 版本](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2022 年 1 月 20 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

另请参阅此发行版的 Amazon IVS [发布说明](#)。

## [R2S3 缩略图配置版本](#)

Amazon IVS 入门更新了[第 3 步：使用可选录制创建通道](#)

2022 年 1 月 18 日

在[自动录制到 Amazon S3](#)中，我们在“录制内容”部分增加了一条有关修改 thumbnails 文件夹的说明，增加了一个新的“缩略图”部分，并在“JSON 元数据文件”中部分更改了有关 thumbnails 和 path 字段的信息。

## [Android 播放器指南](#)

在“安装库”部分删除了 jcenter() 行，因为 jCenter 已被弃用。

2022 年 1 月 7 日

## [iOS 播放器](#)

添加了一个关于播放器在测试 arm64e 架构时崩溃的“已知问题”。

2021 年 12 月 20 日

<a href="#">广播开发工具包 1.2.0 发布</a>	<p>在所有广播开发工具包指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a> 和 <a href="#">iOS</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>上，更新了指向新版本的广播开发工具包参考链接。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2021 年 12 月 9 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	<p>扩充了有关流/会话开始/创建/结束事件的描述内容，并添加了使用建议。</p>	2021 年 12 月 3 日
<a href="#">串流配置</a>	<p>对于来自 Android 和 iOS 的流，将 Larix Broadcast 上的信息替换为指向 Amazon IVS 广播开发工具包文档的指针。</p>	2021 年 11 月 24 日
<a href="#">广播：SDK for Android 指南</a>	<p>添加了有关 Android 5/6/7 设备的问题，这些设备只能使用系统的默认麦克风，因此无法接收广播开发工具包针对麦克风的 onDeviceAdded 和 onDeviceRemoved 回调。</p>	2021 年 11 月 24 日
<a href="#">播放器 1.6 发布</a>	<p>在所有播放器指南中，更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js 集成</a>和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2021 年 11 月 23 日

## [Amazon IVS 播放器](#)

在介绍性文本的末尾，添加了一段关于转换支持的段落以及指向 Amazon IVS 广播开发工具包文档的指针。

2021 年 11 月 23 日

## [监控 Amazon IVS 实时流运行状况](#)

为新 Amazon IVS 功能新增了用户指南页面。对于流运行状况，我们还：

2021 年 11 月 18 日

- 在 [Amazon IVS 入门](#) 的“第 2 步：设置 IAM 权限”中更新了 IAM policy：添加了三个 IVS 权限 ( `GetStream`、`GetStreamSession`、`ListStreamSessions` ) 和 `cloudwatch:GetMetricData`。
- 在“使用亚马逊监控亚马逊 IVS”中添加了四个高分辨率指标 `CloudWatch:IngestAudioBitrate`、`IngestFrameRate`、`IngestVideoBitrate` 和 `KeyframeInterval`。

2023 年 10 月 12 日更新：  
此 CloudWatch 文档已删除，内容已移至 [Monitoring IVS 低延迟流媒体](#)。

- 在“[使用 Amazon 和 Amazon EventBridge IVS](#)”中添加了两个事件：会话已创建和会话已结束。

<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	更新了录制开始事件的描述。	2021 年 11 月 5 日
<a href="#">广播：SDK for iOS 指南</a>	为 AirPods 连接到 iOS 12 设备添加“已知问题”。	2021 年 11 月 4 日
<a href="#">使用 FFmpeg 的流</a>	在流传输配置中，澄清了 FFmpeg 可以用于许多操作系统和设备（不仅仅是 Windows 桌面设备），并修复了网络摄像机项目符号中示例的格式。	2021 年 11 月 3 日
<a href="#">广播开发工具包 ( Android 和 iOS ) 1.1.0 版本</a>	<p>在广播开发工具包指南中，更新了新版本的版本号和构件链接：<a href="#">Android</a> 和 <a href="#">iOS</a>。在 Android 中，“创建广播配置”中存在新的 setPosition 坐标。在 iOS 中，“创建广播配置”中存在一个新的高级使用案例（“使用后台视频”）、插槽位置更改以及新的“已知问题”。</p> <p>在 <a href="#">Amazon IVS 文档登录页</a> 面上，更新了指向新版本的广播开发工具包参考链接。</p> <p>增加了一个新页面“<a href="#">广播：混合器指南</a>”到此功能的文档中。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2021 年 10 月 20 日
<a href="#">设置私有频道</a>	在“令牌架构”中，更新了 access-control-all-ow-origin 定义以引用“源”，而不是“域”。	2021 年 10 月 11 日



<a href="#">Android 播放器 1.5.1 版本</a>	错误修复版本；请参阅 <a href="#">Amazon IVS 发布说明</a> 。还更新了 <a href="#">Android 播放器指南</a> 的链接和文本中的 Android 播放器指南版本号引用。	2021 年 9 月 29 日
<a href="#">播放器 1.5.0 版本</a>	在所有播放器指南中，更新了新版本的版本号和构件链接： <a href="#">Web</a> 、 <a href="#">Android</a> 、 <a href="#">iOS</a> 、 <a href="#">Video.js 集成</a> 和 <a href="#">JW 播放器集成</a> 。  在 <a href="#">Amazon IVS 文档登录页面</a> ，更新了播放器 SDK 参考链接，以指向新版本。  另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a> 。	2021 年 9 月 28 日
<a href="#">串流配置</a>	在“音频设置”中，指定了最小比特率：96 Kbps。	2021 年 9 月 22 日
<a href="#">Amazon IVS 入门</a>	在“步骤 4：设置串流软件”中，添加了关于在 30 秒内未发送数据时断开连接的说明。	2021 年 9 月 20 日
<a href="#">基于身份的策略示例</a>	在 Amazon IVS Security 中，修复了 <a href="#">访问 Amazon IVS 通道</a> ：添加了结束标点符号 (}})} 的示例中的打字错误。	2021 年 9 月 17 日
<a href="#">播放器 1.4.1 和 1.4.0 的软件开发工具包大小</a>	在播放器 <a href="#">1.4.1</a> 和 <a href="#">1.4.0</a> 发布说明中，我们对移动软件开发工具包大小表进行了更正。	2021 年 9 月 16 日

## [播放器 1.4.1 版本](#)

错误修复版本；请参阅 [Amazon IVS 发布说明](#)。同时所有播放器指南中更新了版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。

2021 年 9 月 8 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

在串流配置中，更新了[隐藏式字幕](#)的信息。

## [广播：SDK for Android 指南](#)

在“设置 ImagePreviewView 为预览”中，对文本进行了细微的澄清。在“切换摄像头”中，修复了两个打字错误。在“创建直播配置”中，删除了线路引用 `video.setDefaultAspectMode`，现在不可用。

2021 年 9 月 1 日

## [使用 FFmpeg 进行串流配置](#)

修改了捕获视频文件的设置。具体而言，`-g 120` 已更改为 `-force_key_frames expr:gte(t,n_force_d*2)`。这会导致编码器每 2 秒插入一个关键帧，无论源输入帧速率为何。

2021 年 8 月 23 日

## [Amazon IVS 播放器：适用于 Web 的开发工具包指南](#)

为 Pixel 4/4a 移动浏览器添加了新的“已知问题”。

2021 年 8 月 20 日

## [Amazon IVS 播放器：Video.js 集成](#)

在“示例代码”中，将版本号更新为 7.14.3。在早于 7.14.3 的 Video.js 版本中存在一个安全漏洞。

2021 年 8 月 19 日

<a href="#">串流配置</a>	对于 STANDARD 通道类型，添加了一个注释，即音频只对呈现 360p 及以下的音频进行解码；在此之上的音频将会通过。	2021 年 8 月 18 日
<a href="#">Amazon IVS 入门</a>	在“步骤 2：设置 IAM 权限”中，添加了将策略附加到现有用户的步骤。此新过程是对旧过程的补充，该过程用于创建新用户并将策略附加到该用户。	2021 年 8 月 11 日
<a href="#">播放器 1.4.0 发行版</a>	<p>在所有播放器指南中，更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a>、<a href="#">Video.js 集成</a>和 <a href="#">JW 播放器集成</a>。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>另请参阅此发行版的 Amazon IVS <a href="#">发布说明</a>。</p>	2021 年 8 月 10 日
<a href="#">Amazon IVS 播放器：适用于 Web 的开发工具包指南</a>	在“使用 NPM 设置”中，添加了关于在您自己的域中托管播放器静态资产的注释。	2021 年 7 月 30 日

<a href="#">Amazon IVS 入门</a>	<p>在“步骤 2：设置 IAM 权限”中，更新了策略信息和说明。</p> <p>在“步骤 3：使用可选录制创建通道”中，添加了“自动录制到 S3”部分（替换前一段）。</p> <p>在“步骤 4：设置流传输软件”中，添加了“使用 Amazon IVS 广播开发工具包进行流传输”部分。</p>	2021 年 7 月 29 日
<a href="#">自动记录至 S3</a>	添加了一个新部分， <a href="#">“从私有存储桶中播放录制的内容。”</a> 还更新了此页面的简介。	2021 年 7 月 28 日
<a href="#">Amazon IVS 广播 SDK ( Android 和 iOS )</a>	适用于 Android 和 iOS 的广播开发工具包的初始版本。请参阅“Amazon IVS 广播 SDK”下的文档，这是 Amazon IVS <a href="#">文档登录页面</a> 的新章节。	2021 年 7 月 27 日
<a href="#">Amazon IVS 播放器</a>	更新了 <a href="#">桌面浏览器</a> ，表示 Amazon IVS 播放器 1.3.0 支持新版本的 macOS Safari 浏览器上的超低延迟。	2021 年 7 月 14 日
<a href="#">Amazon IVS 服务限额</a>	对于 PutMetadata 终端节点，增加了每个账户 155 TPS 的限制。	2021 年 6 月 29 日
<a href="#">ivs.rocks</a>	在 Amazon IVS 用户指南 <a href="#">登录页面</a> 中，添加了 ivs.rocks 的链接和简要描述。	2021 年 6 月 25 日

<a href="#">播放器浏览器和平台要求</a>	对于 Amazon IVS 播放器，添加了列出支持的浏览器最新版本的网站链接。	2021 年 6 月 25 日
<a href="#">串流配置</a>	在“通道类型”中，更新了通道类型的定义。对于 STANDARD 通道，分辨率可高达 1080p；对于 BASIC 通道，分辨率为 480p。（以前的定义仅依据垂直分辨率。）	2021 年 6 月 17 日
<a href="#">成本</a>	增加了一个关于成本的新页面。	2021 年 6 月 17 日
<a href="#">Amazon IVS 播放器：SDK for Android 指南</a>	添加了新的“权限”部分。	2021 年 6 月 17 日
<a href="#">支持播放器移动浏览器</a>	在 <a href="#">移动浏览器</a> 中，添加了有关支持 iPados 版 Chrome 和 iPadOS 版 Safari 的信息。	2021 年 6 月 14 日
<a href="#">播放器 SDK 大小</a>	对 <a href="#">Android</a> 和 <a href="#">iOS</a> 播放器开发工具包指南添加了新的“开发工具包大小”部分。	2021 年 6 月 11 日
<a href="#">Amazon IVS 播放器：适用于 Web 的开发工具包指南</a>	在 iOS 移动浏览器上播放内容时添加了两个“已知问题”（利用 <code>player.getQualities()</code> 和 <code>player.getLiveLatency()</code> 调用）。	2021 年 6 月 9 日

<a href="#">支持的区域和服务端点</a>	<p>将支持的区域列表替换为<a href="#">亚马逊科技一般参考中的 Amazon IVS 页面</a>的链接，当添加对新区域的支持时，它会自动更新。在“使用亚马逊监控亚马逊 IVS CloudWatch”页面上进行了更改。</p> <p>2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring <a href="#">IVS 低延迟流媒体</a>。</p>	2021 年 6 月 8 日
<a href="#">Amazon IVS 播放器问题</a>	<p>在 <a href="#">Web</a>、<a href="#">Android</a> 和 <a href="#">iOS</a> 播放器的“已知问题和解决方法”中，要求客户将所有问题报告给支持中心。还添加了 Android 11 仿真器的问题。</p>	2021 年 6 月 4 日
<a href="#">Android 和 iOS 播放器 1.3.3 发行版</a>	<p>错误修复版本；请参阅 Amazon IVS <a href="#">发布说明</a>。还更新了 <a href="#">Android 播放器指南</a> 和 <a href="#">iOS 播放器指南</a> 中链接和文本中的版本号引用。</p> <p><a href="#">Amazon IVS 文档登录页面</a> 始终指向最新版本的播放器开发工具包参考。</p>	2021 年 6 月 1 日
<a href="#">设置私有频道</a>	<p>更新了“生成并签署播放令牌”（有关创建签名的信息和“说明”中的步骤）。</p>	2021 年 5 月 26 日
<a href="#">全球与区域</a>	<p>将“全球解决方案，区域控制”从“Amazon IVS 入门”移至<a href="#">什么是 Amazon IVS</a>。</p>	2021 年 5 月 21 日

<a href="#">Amazon IVS 播放器 : Video.js 集成</a>	在“示例代码”中，将 Cloudflare 版本号从 7.6.6 更新为 7.11.4。	2021 年 5 月 20 日
<a href="#">Android 播放器 1.3.2 发行版</a>	错误修复版本；请参阅 Amazon IVS <a href="#">发布说明</a> 。还更新了 <a href="#">Android 播放器指南</a> 的链接和文本中的 Android 播放器指南版本号引用。	2021 年 5 月 19 日
<a href="#">Amazon IVS Service Quotas</a>	措辞略有改动。已删除有关最大标签数量的信息；此信息已移至“API 参考”。	2021 年 5 月 12 日
<a href="#">Amazon IVS 发布说明</a>	为 Web 播放器 1.3.1 添加了一个注释：1.3.0 NPM 软件包存在，但不起作用。	2021 年 5 月 11 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	已在所有相关示例中将 <code>stream_id</code> 更新为“已净化的”值。	2021 年 5 月 10 日
<a href="#">Amazon IVS 播放器 : 适用于 Web 的开发工具包指南</a>	在 iOS 移动浏览器上播放录制的内容时，对 <code>player.seekTo()</code> 调用添加了一个已知问题和解决方法。	2021 年 5 月 10 日
<a href="#">串流配置</a>	将 Encoder Configuration (编码器配置) 页面重命名为 Streaming Configuration (流传输配置)。	2021 年 5 月 6 日

## [将亚马逊 EventBridge 与亚马逊 IVS 配合使用](#)

在“示例：录制状态更改”中，添加了 recording\_duration\_ms 字段，更改了 recording\_s3\_key\_prefix 字段的示例值，并更改了 recording\_status\_reason 字段的值。

2021 年 5 月 5 日

## [播放器 1.3 发行版](#)

在所有播放器指南中，更新了新版本的版本号和构件链接：[Web](#)、[Android](#)、[iOS](#)、[Video.js 集成](#)和 [JW 播放器集成](#)。对于 Android，已将 mavenCentral() 添加到“安装库”。

2021 年 5 月 5 日

在 [Amazon IVS 文档登录页面](#)，更新了播放器 SDK 参考链接，以指向新版本。

在播放器 1.3.0 及更高版本中，适用于 iOS 的 Chrome 和 Safari 浏览器现在支持定时元数据。这在 [IVS 播放器开发工具包 概述](#)（关于“移动浏览器”的表格）和 [将元数据嵌入视频流中](#)（在“使用元数据”）中标注。

另请参阅 Amazon IVS 对于此发布的[发行说明](#)。

## [Amazon IVS Service Quotas](#)

添加了新章节“Service Quotas 与 CloudWatch 使用量指标集成”。

2021 年 4 月 26 日



<a href="#">流的最长持续时间</a>	在 <a href="#">Amazon IVS 入门</a> ( “第 4 步：设置流软件” ) 中，添加了关于流的最长持续时间 48 小时的说明。	2021 年 4 月 23 日
<a href="#">IAM policy 更改</a>	进行了多项 IAM policy 更改： <ul style="list-style-type: none"><li>• <a href="#">Amazon IVS 入门</a>：在“步骤 2：设置 IAM 权限”中，添加了服务限额。</li><li>• <a href="#">Amazon IVS 安全性</a>：在“使用 Amazon IVS 控制台”中，简化了策略示例并添加了服务限额。</li></ul>	2021 年 4 月 22 日
<a href="#">新 CloudWatch 指标</a>	发布新 CloudWatch 指标时对文档进行了各种更改： <ul style="list-style-type: none"><li>• 使用 Amazon 监控 Amazon IVS CloudWatch — 添加了新指标：并发观看次数和并发流。</li></ul> <p>2023 年 10 月 12 日更新： 此 CloudWatch 文档已删除，内容已移至 <a href="#">Monitoring IVS 低延迟流媒体</a>。</p> <ul style="list-style-type: none"><li>• <a href="#">服务限额</a>：更新了相关限额的名称以匹配新指标。</li><li>• <a href="#">术语表</a>：添加了“视图”。</li></ul>	2021 年 4 月 13 日

## [自动记录至 S3](#)

为这项新 Amazon IVS 功能新增了用户指南页面。这还会影响几个现有文档：

2021 年 4 月 7 日

- [Amazon IVS 入门](#) – 添加了针对 R2S3 的 IAM policy 信息。改写了通道创建步骤。添加了一个在 OBS Studio 中选择性启用本地录制的段落。新增了禁用录制部分。
- [将 Amazon EventBridge 与 Amazon IVS 配合使用](#) — 添加了记录状态更改事件。
- 使用亚马逊监控亚马逊 IVS CloudWatch - 新增 RecordedTime 指标。

2023 年 10 月 12 日更新：  
此 CloudWatch 文档已删除，内容已移至 [Monitoring IVS 低延迟流媒体](#)。

- [Amazon IVS 安全](#) – 添加了关于“为 Amazon IVS 使用服务相关角色 (SLR)”的章节。
- [服务限额](#) – 为新的录制配置端点添加了“API 调用速率限额”，并为录制配置增加了“资源限额”限制。

## [Amazon IVS 串流配置](#)

在 Closed Captioning (隐藏式字幕) 中，阐明了播放器软件开发工具包仅支持 1 种语言，而不支持多轨道字幕播放。

2021 年 3 月 29 日

<a href="#">全球与区域</a>	在 <a href="#">什么是 Amazon IVS</a> 中，添加了“全球解决方案，区域控制”新章节，以说明什么是全球与区域。在 <a href="#">Amazon IVS 入门</a> 中，在创建通道的说明中提到了选择一个区域。	2021 年 3 月 25 日
<a href="#">EventBridge 事件延迟和 IDR/关键帧编码器设置</a>	<p>澄清了某些 EventBridge 事件中 IDR/Keyframe 视频编码器设置与延迟之间的关系。这会影响以下两个文档：</p> <ul style="list-style-type: none"><li>• <a href="#">“Amazon IVS 流式传输配置”</a>：请参阅“减少延迟”中的 IDR/Keyframe 项目列表。</li><li>• <a href="#">“将 Amazon EventBridge 与 Amazon IVS 配合使用”</a> — 参见新的“直播状态更改事件延迟注意事项”。</li></ul>	2021 年 3 月 25 日
<a href="#">使用亚马逊监控亚马逊 IVS CloudWatch</a>	<p>澄清 CloudWatch 了数据保留多长时间。</p> <p>2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 <a href="#">Monitoring IVS 低延迟流媒体</a>。</p>	2021 年 3 月 18 日
<a href="#">串流配置</a>	在 Audio Settings (音频设置) 中，将支持的比特率从 192 更改为 320Kpbs。	2021 年 3 月 15 日

<a href="#">TLS 版本要求</a>	<p>明确了 TLS ( 传输层安全 ) 的要求。对于 API 调用，客户端必须支持 TLS 1.0 或更高版本，但我们建议使用 TLS 1.2 或更高版本。对于流传输/回放，需要 TLS 1.2 版或更高版本。</p> <p>对以下两个文件进行了修改： 流传输配置 ( <a href="#">“流摄取：编解码器、RTMPS 和端口 443”</a> 相关文章 ) 和安全性 ( <a href="#">“基础设施安全性”</a> 相关文章 )。</p>	2021 年 3 月 15 日
<a href="#">Amazon IVS 播放器：适用于 Web 的开发工具包指南</a>	添加了 HTML5 和 setQuality() 的已知问题。	2021 年 3 月 15 日
<a href="#">Amazon IVS 播放器：适用于 Web 的开发工具包指南</a>	添加了字幕已知问题。	2021 年 3 月 11 日
<a href="#">Amazon IVS 播放器</a>	<p>在 <a href="#">SDK for Android 指南</a> 和 <a href="#">SDK for iOS 指南</a> 中添加了关于“线程安全”的章节。</p> <p>此外，对于 Android，请注意在调用该 player.release() 方法后，则无法再使用此播放器。</p>	2021 年 3 月 2 日
<a href="#">使用亚马逊监控亚马逊 IVS CloudWatch</a>	<p>更新了使用 CloudWatch 控制台访问 Amazon IVS 指标的程 序：添加了有关何时列出“IVS”的信息以及屏幕截图。</p> <p>2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring <a href="#">IVS 低延迟流媒体</a>。</p>	2021 年 2 月 26 日

<a href="#">安全性</a>	在“基础设施安全性”中添加了一条说明，即 Amazon IVS 流传输要求 TLS 1.2 版本。还列出了一个有关 AWS 全球网络安全流程的新网页。	2021 年 2 月 17 日
<a href="#">Amazon IVS 播放器：JW 播放器集成</a>	适用于 Amazon IVS 播放器的有关 JW 播放器插件的新用户指南页面。还将 JW 播放器行添加到 Web 播放器指南中的 <a href="#">框架集成表</a> 。	2021 年 1 月 28 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	扩展了关于发送事件保证的措施。	2021 年 1 月 22 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	已添加：将尽最大努力发送事件。	2021 年 1 月 13 日
<a href="#">串流配置</a>	已将编解码器音频设置从 AAC 更改为 AAC (LC)。	2020 年 12 月 18 日
<a href="#">Amazon IVS Service Quotas</a>	在“资源限额”中，添加了资源的最大标签数。	2020 年 12 月 17 日
<a href="#">Android 播放器 1.2.1 发行版</a>	错误修复版本；请参阅 <a href="#">Amazon IVS 发行说明</a> 。还更新了 <a href="#">Android 播放器指南</a> 的链接和文本中的 Android 播放器指南版本号引用。	2020 年 12 月 16 日
<a href="#">Amazon IVS 版本注释</a>	对于 Amazon IVS Android 播放器 1.2.0 和 1.1.0，添加了一个导致开发工具包崩溃的已知问题。	2020 年 12 月 11 日

<a href="#">Amazon IVS 入门</a>	在播放 URL 的项目符号 (在“第 3 步：创建通道”) 中，添加了一个注释，即不支持自定义播放域。	2020 年 12 月 4 日
<a href="#">Amazon IVS 版本注释</a>	删除了 iOS 播放器 1.0.6 和 1.0.0 的下载链接；这些版本已弃用。  为 iOS 播放器 1.2.0 添加了一个“已知问题”。	2020 年 12 月 4 日
<a href="#">播放器 1.2.0 发行版</a>	在所有播放器指南中，更新了新版本的版本号和构件链接： <a href="#">Web</a> 、 <a href="#">Android</a> 、 <a href="#">iOS</a> 以及 <a href="#">Video.js 集成</a> 。在 Android 指南中添加了一个已知问题。  在 <a href="#">Amazon IVS 文档登录页面</a> ，更新了播放器 SDK 参考链接，以指向新版本。  另请参阅更新后的 Amazon IVS <a href="#">发行说明</a> 。	2020 年 11 月 23 日
<a href="#">设置私有频道</a>	在“生成和签名播放令牌”部分，JWT 有效载荷中的 channel-arn 值是一个字符串。	2020 年 11 月 18 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	已将 stream_id 字段添加到许多事件中。这是每次通道上线时分配的唯一标识符。对于给定通道，每个实时流都有一个新的 stream_id。流 ID 允许客户区分同一通道上的不同流会话。	2020 年 11 月 12 日

<a href="#">在视频流中嵌入元数据</a>	从 Amazon IVS 控制台新增了“查看定时元数据”相关章节。	2020 年 11 月 9 日
<a href="#">Web 播放器指南</a>	更新了“内容安全策略”相关章节，特别是在使用 Safari 时将资产托管在单独的页面上。	2020 年 11 月 4 日
<a href="#">服务限额 (CCV 和 CCB 限额)</a>	增加了有关确保充分并发查看者和并发广播者限制的重要性的说明，尤其是在大型流传输事件之前。请参阅 <a href="#">Amazon IVS 入门</a> 和 <a href="#">Amazon IVS Service Quotas</a> 。	2020 年 11 月 4 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	更新了限制违规事件：JSON blob 的 detail 章节针对所有这些事件使用 limit_name。（以前只有并发广播显示这一点，其他广播显示 limit。）	2020 年 10 月 28 日
<a href="#">设置私有频道</a>	在“生成和签名播放令牌”章节中指出，JWT 有效载荷中的 exp（过期）字段是一个整数。	2020 年 10 月 27 日
<a href="#">Amazon IVS Service Quotas</a>	增加了三个限制：通道数量、并发查看器和并发广播。	2020 年 10 月 27 日
<a href="#">Web 播放器 1.1.2 发行版</a>	错误修复版本；请参阅 <a href="#">Amazon IVS 发布说明</a> 。链接和文本中的版本号引用已在 <a href="#">Web 播放器指南</a> 和 <a href="#">Video.js 集成指南</a> 中更新。	2020 年 10 月 9 日

<a href="#">摄取分辨率限额与事件</a>	为采集解析添加了服务配额和 EventBridge 事件。参见 <a href="#">亚马逊 IVS 服务配额</a> 和将 <a href="#">亚马逊 EventBridge 与亚马逊 IVS 配合使用</a> 。	2020 年 10 月 9 日
<a href="#">播放器 1.1.0 发行版</a>	<p>在所有播放器指南中，更新了新版本的版本号和构件链接：<a href="#">Web</a>、<a href="#">Android</a>、<a href="#">iOS</a> 以及 <a href="#">Video.js 集成</a>。</p> <p>在 iOS 和 Web 指南中，添加了一个关于“已知问题”的新章节。</p> <p>在 <a href="#">Amazon IVS 文档登录页面</a>，更新了播放器 SDK 参考链接，以指向新版本。</p> <p>在 <a href="#">Amazon IVS 播放器概述</a>中，删除了 Android getSessionId 函数（尚不可用）。</p>	2020 年 10 月 7 日
<a href="#">设置私有频道</a>	添加了一个新章节，“私有通道的工作流”。在关于生成和签名令牌的章节中，澄清了有效载荷字段的描述和示例。更正了列出和获取播放密钥对的示例。	2020 年 9 月 21 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	在多个事件中添加了 channel_name 字段。	2020 年 9 月 14 日



<a href="#">在视频流中嵌入元数据</a>	扩展了有关设置 IAM 权限 ( 完整程序和策略 )、插入元数据 ( 添加了 CLI 过程 ) 和使用元数据 ( 链接到多个 GitHub 演示 ) 的信息。	2020 年 9 月 14 日
<a href="#">播放器指南</a>	澄清每个播放器的最新版本 ( <a href="#">Web</a> 、 <a href="#">Android</a> 、 <a href="#">iOS</a> 和 <a href="#">Video.js 集成</a> )。	2020 年 9 月 9 日
<a href="#">Amazon IVS 入门</a>	提及在控制台中查看新流之前，有一个短暂的延迟。	2020 年 9 月 9 日
<a href="#">Amazon IVS 版本注释</a>	已将播放器 iOS 下载链接更改为与播放器 iOS 指南中的链接相同。	2020 年 9 月 9 日
<a href="#">在视频流中嵌入元数据</a>	添加了与亚马逊云科技博客文章相关的链接。	2020 年 9 月 3 日
<a href="#">Amazon IVS 播放器</a>	扩展了有关播放器功能的讨论。阐明了只能保证 Amazon IVS 播放器 ( 而非第三方播放器 ) 的性能。	2020 年 9 月 3 日
<a href="#">Amazon IVS Service Quotas</a>	更正了相关问题，指示只能调整通道、并发查看器和并发广播限额。	2020 年 8 月 31 日
<a href="#">串流配置</a>	进行了若干更改，为“避免第三方流传输/转发服务”一章添加了 <a href="#">减少延迟</a> 小节，并阐明了我们强烈建议使用 <a href="#">CBR</a> 而不是 VBR 的原因。	2020 年 8 月 24 日
<a href="#">在视频流中嵌入元数据</a>	更新了 <a href="#">使用定时元数据</a> 中的 Web 示例。	2020 年 8 月 24 日

<a href="#">Amazon IVS 播放器 : SDK for Android 指南</a>	在 <a href="#">安装库</a> 中更新了代码示例。	2020 年 8 月 24 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	在关于“示例：违反限制”部分中，更新了几个字段名称：limit_name、limit_value、exceeded_by 和 limit_unit。这些名称包括下划线（并非破折号）。	2020 年 8 月 19 日
<a href="#">设置私有频道</a>	<p>新增有关新的 Amazon IVS 功能的用户指南页面，支持私有通道。这还会影响几个现有文档：</p> <p><a href="#">Amazon IVS 入门</a>和使用 <a href="#">AWS 记录 Amazon IVS API 调用 CloudTrail</a>：频道新增 authorized 字段。</p> <p><a href="#">安全性</a>：进行了几处更改，包括新增章节“特权和非特权访问”。</p> <p><a href="#">服务限额</a>：添加了多个播放限额。</p> <p><a href="#">术语表</a>：添加了播放密钥对。</p>	2020 年 8 月 19 日
<a href="#">Amazon IVS 入门</a>	增加了有关 <a href="#">亚马逊云科技区域服务</a> 的新章节。	2020 年 8 月 11 日

<a href="#">Amazon IVS 播放器 : SDK for iOS 指南</a>	更新了指向参考文档和框架下载的连接，以访问 1.0.6 版本。同时更新了 Amazon IVS <a href="#">文档登录页面</a> 的参考文档链接。	2020 年 8 月 11 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	Amazon IVS EventBridge 活动现已通过亚马逊 EventBridge 控制台提供。请参阅“为亚马逊 IVS 创建亚马逊 EventBridge 规则”部分。	2020 年 8 月 5 日
<a href="#">Amazon IVS 播放器 : Video.js 集成</a>	在“使用 NPM 设置”章节中，将待安装的 Video.js npm 程序包的链接更新到版本 7.6.6。	2020 年 7 月 30 日
<a href="#">将亚马逊 EventBridge 与亚马逊 IVS 配合使用</a>	对于 Amazon IVS 流状态和流运行状况更改，事件名称将在名为 event_name （不是以前所记录的 eventName ）的字段中提供。	2020 年 7 月 29 日
<a href="#">Amazon IVS 入门</a>	更改了设置流传输软件的说明，以指示 Amazon IVS 接收需要端口 443。这也会影响流传输配置文档；请参阅 <a href="#">RTMPS 和端口 443</a> 的新章节。	2020 年 7 月 27 日
<a href="#">Amazon IVS 播放器 : SDK for iOS 指南</a>	在手动安装框架的说明中，更改了最新版本的下载位置。	2020 年 7 月 27 日
<a href="#">在视频流中嵌入元数据</a>	添加了使用定时元数据相关的 Android 和 iOS 示例。	2020 年 7 月 24 日
<a href="#">新服务和用户指南</a>	这是 Amazon Interactive Video Service (IVS) 的初始版本。	2020 年 7 月 15 日

# IVS Low-Latency Streaming API Reference 更改

API 更改	描述	日期
无令牌播放限制	<ul style="list-style-type: none"> <li>• 添加了新资源 PlaybackRestrictionPolicy。</li> <li>• 添加了五个 PlaybackRestrictionPolicy ( ies ) 端点 ( 创建/删除/获取/列出/更新 )。</li> <li>• 添加了 PlaybackRestrictionPolicy 和 PlaybackRestrictionPolicySummary 对象。</li> <li>• 已 playbackRestrictionPolicyArn 添加到频道和 ChannelSummary 物体中。这会影 响频道端点响应 ( 创建/ BatchGet /获取/列表/更新 )。</li> <li>• 在 ListChannel 请求中添加了 filterByPlaybackRestrictionPolicyArn 。</li> </ul>	2024 年 1 月 31 日
通道类型定义	更新了通道类型定义以提供更多详细信息，尤其是有关渲染的转码阶梯。请参阅 IVS Low-Latency Streaming API Reference 中的 <a href="#">Channel Types</a> ( 通道类型 )。	2023 年 8 月 18 日
R2S3 rendition filtering and thumbnail enhancements	<ul style="list-style-type: none"> <li>• 在 ThumbnailConfiguration、添加 resolution 和 storage。这会影 响 CreateRecordingConfiguration 请求和响 应、 GetRecordingConfiguration 响应和 GetStreamSession 响应。</li> <li>• 在中 ThumbnailConfiguration ， 将 targetIntervalSeconds 最小值从 5 更改为 1，并更新了“重要”注释使其仅适用于 BASIC 频道。</li> <li>• 已添加 RenditionConfiguration 对象。</li> <li>• 已 renditionConfiguration 添加到 RecordingConfiguration 对象中。这会影 响三个响应： CreateRecordingConfiguration GetRecordingConfiguration、和 GetStream</li> </ul>	2023 年 7 月 17 日

API 更改	描述	日期
	Session。我们还在 CreateRecordingConfiguration 请求中 renditionConfiguration 添加了内容。	
私有通道的查看者会话撤消	<ul style="list-style-type: none"> <li>• 增加了两个端点：StartViewerSession Revocation 和 BatchStartViewerSessionRevocation。</li> <li>• 增加了两个对象：BatchStartViewerSessionRevocationError 和 BatchStartViewerSessionRevocationViewerSession。</li> </ul>	2023 年 6 月 28 日
高级通道类型	<ul style="list-style-type: none"> <li>• 增加了新的通道 type 值和定义。这会影 响两个请求 ( Create/UpdateChannel ) 和 Channel 对象。</li> <li>• 已将该preset字段添加到频道和 ChannelSummary 物体中。这会影 响多个请求 (Create/UpdateChannel) 和响应 (BatchGetChannel , Create/Get/ UpdateChannel、 )。 GetStream Session ListChannels</li> <li>• 已type添加到 ChannelSummary 对象中。这会影 响 ListChannels 响应。</li> </ul>	2023 年 6 月 2 日
RTMP 支持	已将该insecureIngest 字段添加到频道和 ChannelSummary 物体中。这会影 响多个请求和 响应。	2023 年 3 月 30 日
流状态	在 Stream and ob StreamSummary jects 中 , 注意不应依赖该state字段的OFFLINE值。相反, “NotBroadcasting” 错误将表示直播未直播。	2023 年 2 月 8 日

API 更改	描述	日期
合并片段化的流	在 <code>CreateRecordingConfiguration</code> 请求和 <code>RecordingConfiguration</code> 对象中添加了 <code>recordingReconnectWindowSeconds</code> 字段。这会影响三个响应 ( <code>CreateRecordingConfiguration</code> 、 <code>GetRecordingConfiguration</code> 、和 <code>GetStreamSession</code> )。	2022 年 8 月 30 日
将 BASIC 通道扩展到 1080p	更新了 <code>CreateChannel</code> 、 <code>UpdateChannel</code> 、和 <code>Channel</code> 对象中的频道 <code>type</code> 定义。	2022 年 8 月 16 日
有关 tags 的限制	更新了有关 tags 限制的信息。除了我们链接的亚马逊云科技文档中载明的外，Amazon IVS 对标签没有任何限制。这会影响“欢迎”部分以及多个端点和数据类型。	2022 年 8 月 12 日
<code>maxResults</code> 的最大值和默认值	更新了 <code>maxResults</code> 的最大值和默认值，以反映系统的实际行为。影响所有使用 <code>maxResults</code> 的 List 端点。	2022 年 8 月 12 日
时间戳字段	为 ISO 8601 字段增加了一个注释，说明这些字段将返回为字符串。由于是自动生成的问题，这些会在我们记录的语法中显示为 <code>number</code> 。	2022 年 3 月 28 日
API 授权	在“身份验证与授权”中，阐述了授权的要点。	2022 年 3 月 18 日
标签端点中的 ARN 编码	对于三个标签端点，添加了一条声明，说明 <code>resourceArn</code> 字段必须是 URL 编码字段。	2022 年 3 月 18 日
音频/视频配置对象	更新了 <code>AudioConfiguration</code> 和 <code>VideoConfiguration</code> 数据类型的定义，以表明它们用于监控。（配置在广播公司的编码器中完成。）	2022 年 2 月 17 日

API 更改	描述	日期
R2S3 缩略图配置版本	<p>在 RecordingConfiguration 对象中添加了一个新字段 (thumbnailConfiguration )。这反过来又会影响 CreateRecordingConfiguration 请求和响应、 GetRecordingConfiguration 响应和 GetStreamSession 响应。</p> <p>添加了一个新对象: ThumbnailConfiguration.</p>	2022 年 1 月 18 日
流运行状况发布	<p>添加了 2 个端点 : GetStreamSession 和 ListStreamSessions。</p> <p>添加了 7 个对象 : AudioConfiguration IngestConfiguration、 StreamEvent、 StreamFilters、 StreamSession、 StreamSessionSummary、 和 VideoConfiguration。</p> <p>已将该streamID字段添加到直播和 StreamSummary 对象中。这反过来又会影响 GetStream 和的 ListStreams 响应。</p> <p>已将该filtersBy 字段添加到 ListStreams 请求中。</p>	2021 年 11 月 18 日
时间字段的格式	更新了 Stream 和 StreamSummary 对象 startTime 中的描述 , 添加了以字符串形式返回的 ISO 8601 时间戳。	2021 年 9 月 21 日
STANDARD 通道类型	对于 STANDARD 通道类型 , 添加了注释 , 即只对呈现 360p 及以下的音频进行解码 ; 在此之上的音频将会通过。	2021 年 8 月 18 日
ListTagsForResource 端点	删除了对分页的支持 ; 即 maxResults 请求字段和 nextToken 请求/响应字段。( 分页无法正常工作。 )	2021 年 8 月 13 日
PutMetadata 每个账户的 TPS 限额	对于 PutMetadata 终端节点 , 增加了每个账户 155 TPS 的限制。	2021 年 6 月 29 日

API 更改	描述	日期
通道类型定义	更新了通道类型的定义。对于 STANDARD 通道，分辨率可高达 1080p；对于 BASIC 通道，分辨率为 480p。（以前的定义仅依据垂直分辨率。）	2021 年 6 月 17 日
支持的区域和服务终端节点	将支持的区域列表替换为 <a href="#">亚马逊云科技一般参考中的 Amazon IVS 页面</a> 的链接，当添加对新区域的支持时，它会自动更新。在“欢迎”页面上进行了更改。	2021 年 6 月 8 日
标记	在“标记”（在“欢迎”部分）中，添加了可应用于资源的最大标签数 (50)。	2021 年 5 月 12 日
新 CloudWatch 指标	更改了 Stream 和 StreamSummary 对象 viewerCount 中的定义。	2021 年 4 月 13 日
自动记录至 S3	<ul style="list-style-type: none"> <li>• 添加了 4 个录制配置端点（创建、删除、获取、列出）。</li> <li>• 添加 4 种数据类型（DestinationConfiguration、RecordingConfiguration、RecordingConfigurationSummary、S3DestinationConfiguration）。</li> <li>• 在频道以及 ChannelSummary 对象和频道端点中添加了一个 RecordingConfigurationArn 字段。</li> <li>• 已修改 ListChannels 为按录制配置 ARN 进行过滤。</li> </ul>	2021 年 4 月 7 日
身份验证与授权	<ul style="list-style-type: none"> <li>• 添加了“<a href="#">身份验证与授权</a>”章节来阐明这些概念之间的区别。</li> <li>• authorized 字段的说明（在通道数据类型和通道端点中）更改为：“通道是否为私有通道（启用播放授权）。”</li> </ul>	2021 年 3 月 16 日



API 更改	描述	日期
PutMetadata	为 metadata 请求字段添加了一个最小长度 (1)。	2021 年 3 月 4 日
通道延迟模式	在 Create/ UpdateChannel 和 Channel/ ChannelSummary 对象中，添加了对值的描述。latencyMode	2020 年 12 月 18 日
通道默认值	<ul style="list-style-type: none"> <li>在通道数据类型中，为 authorized 添加默认值。</li> <li>在频道数据类型和中 CreateChannel，添加的默认值type。</li> </ul>	2020 年 12 月 17 日
所有端点列表	表示 maxResults 请求字段具有默认值 50。	2020 年 12 月 5 日
直播和 StreamSummary 对象	更改了 viewerCount 字段的描述以指示值 -1 表示请求超时；在这种情况下，请重试。	2020 年 11 月 10 日
身份验证	添加了 Sigv4 签名信息。请参阅 <a href="#">欢迎使用</a> 章节中的“身份验证”。	2020 年 10 月 9 日
DeleteChannel & DeleteStreamKey	已将 HTTP 响应从 200 改为 204。	2020 年 8 月 26 日
DeleteChannel	阐明了如何删除实时通道，以避免错误。	2020 年 8 月 20 日
播放授权（适用于私人通道）	<ul style="list-style-type: none"> <li>新 PlaybackKeyPair 端点</li> <li>频道中的一个新authorized 字段和 ChannelSummary物体</li> <li>新对象， PlaybackKeyPair 以及 PlaybackKeyPairSummary</li> </ul>	2020 年 8 月 19 日
新服务和 <a href="#">API 参考</a>	这是 Amazon Interactive Video Service (IVS) 的初始版本。	2020 年 7 月 15 日

## 舞台 API 参考更改

API 更改	描述	日期
IVS 实时流式传输发布	<p>此发行版附带主要的文档更改。我们将之前的文档重命名为 IVS Low-Latency Streaming，并发布了新的 IVS Real-Time Streaming 文档。<a href="#">IVS 文档登录页面</a> 现在在单独的实时流式传输和低延迟流式传输部分。每个部分都有各自的用户指南和 API 参考。</p> <p>Stage API Reference 是 IVS 实时流式传输文档的一部分，已更名为 <a href="#">IVS Real-Time Streaming API Reference</a>。我们将在 <a href="#">Document History ( Real-Time Streaming )</a> 中列出此 API 参考的未来更新，而不是此处。</p>	2023 年 8 月 7 日
舞台运行状况	<ul style="list-style-type: none"> <li>• 添加了五个终端节点：GetParticipant ListParticipants、GetStageSession、ListStageSessions、ListParticipantEvents。</li> <li>• 添加了五个对象：事件、参与者 ParticipantSummary、StageSession、StageSessionSummary。</li> </ul>	2023 年 5 月 11 日
新功能	这是舞台 API 的初始版本。我们在文档登录页面添加了 <a href="#">舞台 API 参考</a> 图块。	2023 年 3 月 23 日

## IVS Chat API 文档更改

API 更改	描述	日期
新的错误消息	已将 ConflictException 错误添加到 UpdateLoggingConfiguration。	2023 年 3 月 17 日
maximumMessageRate PerSecond 最大值	将 maximumMessageRatePerSecond 的最大值从 10 更改为 100。这会影响到 CreateRoom GetRoom、和 UpdateRoom 端点。	2023 年 1 月 30 日

API 更改	描述	日期
事件 ( 订阅 ) MessageID 字段	在《聊天功能消息收发 API 参考》中的 <a href="#">事件 ( 订阅 )</a> 中，向 Attributes 添加了一个已弃用字段 (MessageID )。包含此内容以实现向后兼容性。	2023 年 1 月 25 日
新的错误类型	添加了 DeleteMessage、 DisconnectUser、 & 的 PendingVerification 错误 SendEvent。	2022 年 12 月 5 日
WebSocket 错误	在 <a href="#">聊天消息 API 参考</a> 中，我们更新了对两个 WebSocket 错误 ( 未经授权和禁止 ) 的描述。	2022 年 11 月 18 日
聊天功能日志记录	<p>该项新功能的初始版本。我们添加了以下内容：</p> <ul style="list-style-type: none"> <li>一种新的资源， LoggingConfiguration</li> <li>五种数据类型： CloudWatchDestinationConfiguration、 DestinationConfiguration、 KinesisDestinationConfiguration、 LoggingConfigurationSummary、 S3 DestinationConfiguration</li> <li>五个终端节点：创建/删除/获取/列 UpdateLoggingConfiguration 出/ (s)</li> <li>RoomSummary 对象的 loggingConfigurationIdentifiers 字段和房间请求/响应</li> </ul>	2022 年 11 月 17 日
CreateChatToken 描述	更新了的描述 CreateChatToken，包括有关使用该 attributes 字段的新指南。	2022 年 11 月 17 日
有关 tags 的限制	更新了有关 tags 限制的信息。除了我们链接的亚马逊云科技文档中载明的外，Amazon IVS Chat 对标签没有任何限制。这会影“欢迎”部分、四个端点 ( CreateRoom ListTagsForResource TagResource、 、 、 UntagResource ) 和 RoomSummary 数据类型。	2022 年 8 月 12 日

API 更改	描述	日期
新功能	<p>这是 Amazon IVS Chat 的初始版本。我们在文档登陆页面添加了《Amazon IVS Chat》章节，其中包含两个 API 参考文档：</p> <ul style="list-style-type: none"><li>• <a href="#">Chat API 参考</a> – 控制面板 API ( HTTPS )</li><li>• <a href="#">聊天消息 API 参考</a> ——数据平面 API ( ) WebSocket</li></ul>	2022 年 4 月 26 日

## 发布说明 ( 低延迟直播功能 )

2024年2月1日

亚马逊 IVS Broadcast SDK : 安卓 1.14.1、iOS 1.14.1、Web 1.8.0 ( 低延迟直播 )

平台	下载和更改
<a href="#">网络广播软件开发工具包 1.8.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/</a></p> <ul style="list-style-type: none"> <li>无更改。</li> </ul>
<a href="#">安卓广播 SDK 1.14.1</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/android/</a></p> <ul style="list-style-type: none"> <li>少量错误修复和改进。</li> </ul>
<a href="#">iOS 广播软件开发工具包 1.14.1</a>	<p><a href="https://broadcast.live-video.net/1.14.1/AmazonIVSBroadcast.xcf-ramework.zip">下载进行低延迟直播 : https://broadcast.live-video.net/1.14.1/AmazonIVSBroadcast.xcf-ramework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/ios/</a></p> <ul style="list-style-type: none"> <li>修复了 iOS 12 上的多个稳定性问题。</li> </ul>

广播开发工具包大小 : Android

架构	压缩大小	未压缩大小
arm64-v8a	1.663MB	4.708 MB
armeabi-v7a	1.482 MB	3.350 MB
x86_64	1.804MB	5.246 MB

架构	压缩大小	未压缩大小		
x86	1.793 MB	4.973 MB		

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小		
arm64	1.640 MB	4.010 MB		

## 2024 年 1 月 31 日

### 无令牌播放限制

此版本允许在播放授权之外进行源站执法和地理围栏。《IVS 低延迟直播用户指南》和《API 参考》已更新；有关更改的详细信息，请参阅[文档历史记录](#)。

## 2024 年 1 月 25 日

### 纯音频播放

IVS 现在完全支持纯音频播放。[请参阅 IVS Player 概述中的纯音频播放和 IVS Web Player 指南中的纯音频播放。](#)

## 2024 年 1 月 18 日

### 亚马逊 IVS Player SDK 1.24.0

平台	下载和更改
<a href="#">网络播放器 1.24.0 和 Video.js 集成以及 JW 播放器集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产：<a href="https://player.live-video.net/1.24.0/amazon-ivs-player.min.js">https://player.live-video.net/1.24.0/amazon-ivs-player.min.js</a></p>

平台	下载和更改
	<p>Video.js 科技资产：<a href="https://player.live-video.net/1.24.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.24.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.24.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.24.0/web/</a></p> <ul style="list-style-type: none"> <li>增加了对纯音频播放的支持。必须使用手动选择纯音频质量<code>setQuality()</code>；在 auto 画质模式下不会自动选择音质。请参阅 Player Web 指南中的<a href="#">纯音频播放</a>。</li> </ul>
<a href="#">安卓播放器 1.24.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.24.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.24.0/android/</a></p> <ul style="list-style-type: none"> <li>没有变化</li> </ul>
<a href="#">iOS Player 1.24.0</a>	<p>下载：<a href="https://player.live-video.net/1.24.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.24.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.24.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.24.0/ios/</a></p> <ul style="list-style-type: none"> <li>没有变化</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.006 MB	2.846 MB
armeabi-v7a	0.88 MB	1.995 MB
x86_64	1.085 MB	2.916 MB
x86	1.058MB	2.982 MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.43 MB	1.08 MB

2024 年 1 月 3 日

亚马逊 IVS Broadcast SDK：安卓 1.13.4、iOS 1.13.4、Web 1.7.0 ( 低延迟直播 )

平台	下载和更改
<a href="#">网络广播软件开发工具包 1.7.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/</a></p> <ul style="list-style-type: none"><li>此低延迟 SDK 没有更改。</li></ul>
<a href="#">安卓广播 SDK 1.13.4</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/android/</a></p> <ul style="list-style-type: none"><li>此低延迟 SDK 没有更改。</li></ul>
<a href="#">iOS 广播软件开发工具包 1.13.4</a>	<p><a href="https://broadcast.live-video.net/1.13.4/AmazonIVSBroadcast.xcf">下载进行低延迟直播</a>：<a href="https://broadcast.live-video.net/1.13.4/AmazonIVSBroadcast.xcf">https://broadcast.live-video.net/1.13.4/AmazonIVSBroadcast.xcf</a> <a href="#">framework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/ios/</a></p> <ul style="list-style-type: none"><li>此低延迟 SDK 没有更改。</li></ul>



## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小		
arm64-v8a	1.663MB	4.704MB		
armeabi-v7a	1.484MB	3.352MB		
x86_64	1.804MB	5.243MB		
x86	1.795MB	4.97 MB		

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小		
arm64	1.63MB	4.01MB		

2023 年 12 月 4 日

## Amazon IVS 广播 SDK：Android 1.13.2 和 iOS 1.13.2（低延迟直播）

平台	下载和更改
<a href="#">Android 广播 SDK 1.13.2</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/android/</a></p> <ul style="list-style-type: none"> <li>此低延迟 SDK 没有更改。</li> </ul>
<a href="#">iOS 广播 SDK 1.13.2</a>	<p>下载进行低延迟直播：<a href="https://broadcast.live-video.net/1.13.2/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.13.2/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/ios/</a></p> <ul style="list-style-type: none"> <li>此低延迟 SDK 没有更改。</li> </ul>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小		
arm64-v8a	1.663MB	4.704MB		
armeabi-v7a	1.484MB	3.352MB		
x86_64	1.804MB	5.243MB		
x86	1.795MB	4.970MB		

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小		
arm64	1.63MB	4.01MB		

2023 年 11 月 21 日

## Amazon IVS 广播 SDK：Android 1.13.1（低延迟直播）

平台	下载和更改
<a href="#">Android 广播 SDK 1.13.1</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.1/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.1/android/</a> <ul style="list-style-type: none"> <li>此低延迟 SDK 没有更改。</li> </ul>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小		
arm64-v8a	1.663MB	4.705MB		

架构	压缩大小	未压缩大小		
armeabi-v7a	1.484MB	3.352MB		
x86_64	1.804MB	5.243MB		
x86	1.795MB	4.971MB		

2023 年 11 月 17 日

## Amazon IVS 广播 SDK : Android 1.13.0 和 iOS 1.13.0 ( 低延迟直播 )

平台	下载和更改
所有移动设备 ( Android 和 iOS )	<ul style="list-style-type: none"> <li>通过采用 RFC 6555“Happy Eyeballs”并添加 BroadcastConfiguration.network.useIPv6 配置选项来启用或禁用 IPv6 进行广播，改进了 IPv6 支持。</li> </ul>
<a href="#">Android 广播 SDK 1.13.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/android/</a></p> <ul style="list-style-type: none"> <li>修复了在释放物体后使用 AudioSource 物体时发生的崩溃问题。</li> <li>增加了对基于 Surfaceview 的预览的支持，以提高性能。Session 和 StageStream 中的现有 getPreview 方法将继续返回 TextureView 的子类，但在未来的 SDK 版本中，这可能会有变化。</li> <li>特别地，如果您的应用程序依赖于 TextureView，则无需更改即可继续。您也可以从 getPreview 切换到 getPreviewTextureView，为默认 getPreview 返回的内容的最终更改做好准备。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>如果您的应用程序没有特别要求 <code>TextureView</code>，我们建议切换到 <code>getPreviewSurfaceView</code> 以降低 CPU 和内存使用率。</li> <li>SDK 现在实现了一种名为 <code>ImagePreviewSurfaceTarget</code> 的新型预览，适合与应用程序提供的 Android Surface 对象共用。它不是 <code>Android View</code> 的子类，具有更高的灵活性。</li> </ul>
<a href="#">iOS 广播 SDK 1.13.0</a>	<p><a href="https://broadcast.live-video.net/1.13.0/AmazonIVSBroadcast.xcf">下载进行低延迟直播</a> : <a href="https://broadcast.live-video.net/1.13.0/AmazonIVSBroadcast.xcf">https://broadcast.live-video.net/1.13.0/AmazonIVSBroadcast.xcf</a>  <a href="#">framework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/ios/</a></p> <ul style="list-style-type: none"> <li>此版本没有任何变化。</li> </ul>

## 广播开发工具包大小 : Android

架构	压缩大小	未压缩大小		
arm64-v8a	1.66MB	4.70MB		
armeabi-v7a	1.48MB	3.35MB		
x86_64	1.80MB	5.24MB		
x86	1.79MB	4.96MB		

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小		
arm64	1.63MB	4.01MB		

2023 年 11 月 14 日

## Amazon IVS 播放器 SDK 1.23.0

平台	下载和更改
<a href="#">Web 播放器 1.23.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p>下载和更改</p> <p>NPM Package : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.23.0/amazon-ivs-player.min.js">https://player.live-video.net/1.23.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.23.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.23.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.23.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.23.0/web/</a></p> <ul style="list-style-type: none"> <li>在 iOS Safari 浏览器中增加了对低延迟播放的支持。</li> </ul>
<a href="#">Android 播放器 1.23.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.23.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.23.0/android/</a></p> <ul style="list-style-type: none"> <li>使用新 UI 和更多细节更新了参考文档。</li> </ul>
<a href="#">iOS 播放器 1.23.0</a>	<p>下载 : <a href="https://player.live-video.net/1.23.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.23.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.23.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.23.0/ios/</a></p>

平台	下载和更改
	<ul style="list-style-type: none"> <li>无更改。</li> </ul>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	0.975MB	2.744MB
armeabi-v7a	0.853MB	1.917MB
x86_64	1.028MB	2.873MB
x86	1.055MB	2.811MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.39MB	0.93MB

2023 年 10 月 16 日

### Amazon IVS 广播 SDK：Web 1.6.0 (低延迟直播)

平台	下载和更改
<a href="#">Web 广播 SDK 1.6.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a> <ul style="list-style-type: none"> <li>此低延迟 SDK 没有更改。</li> </ul>

## 2023 年 10 月 12 日

### Amazon IVS 广播 SDK : Android 1.12.1 ( 低延迟直播 )

平台	下载和更改
<a href="#">Android 广播 SDK 1.12.1</a>	参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.12.1/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.12.1/android/</a> <ul style="list-style-type: none"> <li>修复了调用 <code>BroadcastSession.setListener</code> 会导致错误的问题。</li> </ul>

### 广播开发工具包大小 : Android

架构	压缩大小	未压缩大小
arm64-v8a	1.685MB	5.046MB
armeabi-v7a	1.503MB	3.702MB
x86_64	1.826MB	5.576MB
x86	1.822MB	5.290MB

## 2023 年 10 月 3 日

### Amazon IVS 播放器 SDK 1.22.0

平台	下载和更改
<a href="#">Web 播放器 1.22.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	NPM Package : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a> 脚本资产 : <a href="https://player.live-video.net/1.22.0/amazon-ivs-player.min.js">https://player.live-video.net/1.22.0/amazon-ivs-player.min.js</a>

平台	下载和更改
	<p>Video.js 科技资产：<a href="https://player.live-video.net/1.22.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.22.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.22.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.22.0/web/</a></p> <ul style="list-style-type: none"> <li>添加获取 SDK 版本 <code>MediaPlayerPackage.getVersion()</code> 的静态方法。</li> </ul>
<p><a href="#">Android 播放器 1.22.0</a></p>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.22.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.22.0/android/</a></p> <ul style="list-style-type: none"> <li>在 <code>Player</code> 界面中添加 <code>setNetworkRecoveryMode</code> 功能，用于在网络中断后设置所需的播放行为。</li> </ul>
<p><a href="#">iOS 播放器 1.22.0</a></p>	<p>下载：<a href="https://player.live-video.net/1.22.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.22.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.22.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.22.0/ios/</a></p> <ul style="list-style-type: none"> <li>在 <code>IVSPlayer</code> 界面中添加 <code>setNetworkRecoveryMode</code> 功能，用于在网络中断后设置所需的播放行为。</li> <li>在播放器播放时，现在可使用 <code>IVSPlayerLayer</code> 上的 <code>copyDisplayedPixelBuffer</code> 方法。以前，只有当播放器处于空闲状态时才能调用此方法。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	0.948 MB	2.676 MB



架构	压缩大小	未压缩大小
armeabi-v7a	0.828 MB	1.865 MB
x86_64	1.025 MB	2.741 MB
x86	1.000MB	2.802 MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.37 MB	0.89 MB

2023 年 10 月 2 日

### 控制台内流式传输

现在，您可以从 IVS 控制台进行流式传输。在低延迟流式传输入门中，请参阅[步骤 5：设置流式传输软件](#)。

2023 年 9 月 14 日

### Amazon IVS 广播 SDK：Web 1.5.2 ( 低延迟流式传输 )

平台	下载和更改
<a href="#">Web 广播 SDK 1.5.2</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a>

# 2023 年 8 月 23 日

## Amazon IVS 广播 SDK : Web 1.5.1、安卓 1.12.0 和 iOS 1.12.0 ( 低延迟流式传输 )

平台	下载和更改
<a href="#">Web 广播 SDK 1.5.1</a>	参考文档 : <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a>
<a href="#">Android 广播 SDK 1.12.0</a>	参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/android/</a> <ul style="list-style-type: none"> <li>修复了一个罕见的错误，该错误会导致广播过早结束，并显示消息“在收到来自对方的关闭后已尝试恢复”。</li> </ul>
<a href="#">iOS 广播 SDK 1.12.0</a>	下载进行低延迟直播 : <a href="https://broadcast.live-video.net/1.12.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.12.0/AmazonIVSBroadcast.xcframework.zip</a> 参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/ios/</a> <ul style="list-style-type: none"> <li>更正了 IVSDeviceDiscovery.createAudioSourceWithName 的签名以返回 IVSCustomAudioSource 而不是 IVSCustomImageSource。</li> </ul>

广播开发工具包大小 : Android

架构	压缩大小	未压缩大小
arm64-v8a	1.685MB	5.046MB
armeabi-v7a	1.503MB	3.702MB

架构	压缩大小	未压缩大小		
x86_64	1.826MB	5.576MB		
x86	1.822MB	5.290MB		

### 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小		
arm64	1.58MB	3.88MB		

2023 年 8 月 23 日

### Amazon IVS 广播 SDK：Android 1.7.6 (低延迟流式传输)

平台	下载和更改
<a href="#">Android 广播 SDK 1.7.6</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.6/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.6/android/</a></p> <ul style="list-style-type: none"> <li>修复了一个罕见的错误，该错误会导致广播过早结束，并显示消息“在收到来自对方的关闭后已尝试恢复”。</li> </ul>

### 广播开发工具包大小：Android

架构	压缩大小	未压缩大小		
arm64-v8a	1.630MB	4.689MB		
armeabi-v7a	1.520MB	3.792MB		
x86_64	1.761MB	4.748MB		

架构	压缩大小	未压缩大小		
x86	1.825MB	5.219MB		

2023 年 8 月 22 日

## Amazon IVS 回放器 SDK 1.21.0

平台	下载和更改
<a href="#">Web 播放器 1.21.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><b>NPM Package</b> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产</b> : <a href="https://player.live-video.net/1.21.0/amazon-ivs-player.min.js">https://player.live-video.net/1.21.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产</b> : <a href="https://player.live-video.net/1.21.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.21.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.21.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.21.0/web/</a></p>
<a href="#">Android 播放器 1.21.0</a>	<p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.21.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.21.0/android/</a></p>
<a href="#">iOS 播放器 1.21.0</a>	<p><b>下载</b> : <a href="https://player.live-video.net/1.21.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.21.0/AmazonIVSPlayer.xcframework.zip</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.21.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.21.0/ios/</a></p> <ul style="list-style-type: none"> <li>通过 <code>IVSPlayerLayer</code> 类的全新 <code>copyDisplayedPixelBuffer</code> 方法，添加了对获取最近显示的视频帧的支持。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	0.942MB	2.662MB
armeabi-v7a	0.823MB	1.853MB
x86_64	1.020MB	2.726MB
x86	0.993MB	2.788MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.36MB	0.87MB

2023 年 8 月 7 日

## Amazon IVS 广播 SDK：Web 1.5.0、Android 1.11.0 和 iOS 1.11.0

平台	下载和更改
<a href="#">Web 广播 SDK 1.5.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a> <ul style="list-style-type: none"> <li>修复了一个问题：Safari 中争用情况定期导致媒体曲目检索出错</li> </ul>
<a href="#">Android 广播 SDK 1.11.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/android">https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/android</a>
<a href="#">iOS 广播 SDK 1.11.0</a>	下载低延迟流式传输： <a href="https://broadcast.live-video.net/1.11.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.11.0/AmazonIVSBroadcast.xcframework.zip</a>

平台	下载和更改
	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/ios">https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/ios</a>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.659MB	4.918 MB
armeabi-v7a	1.482 MB	3.590 MB
x86_64	1.804 MB	5.444 MB
x86	1.795 MB	5.160 MB

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.771 MB	1.879 MB

2023 年 7 月 17 日

## R2S3 渲染筛选和缩略图增强功能

现在，IVS 客户可以控制录制到 Amazon S3 时所生成的流渲染，以及生成的缩略图分辨率。有关更多信息，请参阅：

- [Getting Started with IVS](#)：在“Step 4: Create a Channel”>“Console Instructions”中，更新了屏幕截图和说明。
- [Auto-Record to Amazon S3](#)：在“JSON Metadata Files”中，添加了 latest\_thumbnail 并更新了 thumbnail。在“Thumbnails”和“Discovering the Renditions of a Recording”中，添加了渲染分辨率的描述。

- [Costs](#) : 在“Storing Recorded Video”中更新了屏幕截图。
- [IVS API Reference](#) :
  - 在中 ThumbnailConfiguration , 我们添加了resolution和storage。这会影 响 CreateRecordingConfiguration 请求和响应、 GetRecordingConfiguration 响应和 GetStreamSession响应。
  - 在中 ThumbnailConfiguration , 我们将targetIntervalSeconds最小值从 5 更改为 1 , 并更新 了“重要”注释使其仅适用于BASIC频道。
  - 我们添加了 RenditionConfiguration 对象。
  - 我们renditionConfiguration添加到 RecordingConfiguration 对象中。这会影 响三个响应 : CreateRecordingConfiguration GetRecordingConfiguration、 和 GetStreamSession。我们还在 CreateRecordingConfiguration 请求中renditionConfiguration添加了内容。

2023 年 7 月 14 日

## Amazon IVS 回放器 SDK 1.20.0

平台	下载和更改
<a href="#">Web 播放器 1.20.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播 放器集成</a>	<p><b>NPM Package</b> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产</b> : <a href="https://player.live-video.net/1.20.0/amazon-ivs-player.min.js">https://player.live-video.net/1.20.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产</b> : <a href="https://player.live-video.net/1.20.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.20.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.20.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.20.0/web/</a></p> <ul style="list-style-type: none"> <li>• 修复了在 iOS 移动浏览器上播放实时流或录 制的内容时，播放器 getLiveLatency() 调用返回 0 的问题。(从 Web 播放器 1.17.0 开始，此问题已得到修复。)</li> <li>• 修复了 amazon-ivs-player npm 包的类 型定义。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>在 <a href="#">Web 播放器 SDK 参考</a> 中，添加了新的登录页面并删除了重复的条目。</li> <li>添加对 Video.js 版本 8+ 的支持。</li> </ul>
<a href="#">Android 播放器 1.20.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.20.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.20.0/android/</a>
<a href="#">iOS 播放器 1.20.0</a>	下载： <a href="https://player.live-video.net/1.20.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.20.0/AmazonIVSPlayer.xcframework.zip</a>  参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.20.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.20.0/ios/</a>  <ul style="list-style-type: none"> <li>iOS SDK 现在需要 iOS 12.0 或更高版本（iOS 11 不再受支持。）</li> </ul>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.039 MB	2.922 MB
armeabi-v7a	0.909MB	2.043 MB
x86_64	1.094 MB	3.069 MB
x86	1.126 MB	3.006 MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB



# 2023 年 7 月 13 日

## Amazon IVS 广播 SDK : Web 1.4.0、Android 1.10.0 和 iOS 1.10.0

平台	下载和更改
<a href="#">Web 广播 SDK 1.4.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"><li>修复了以下错误：SDK 提供的键入信息不足以供主机应用程序使用。</li><li>修复了以下错误：即便已经离开，<code>leave()</code> 和后续 <code>refreshStrategy()</code> 也可以重新发布媒体。</li><li>修复了以下错误：刷新策略时，<code>stageStreamsToPublish</code> 返回单个曲目（音频或视频）可能会导致无法进行干净更新。</li><li>添加了关闭浏览器选项卡时更快的断开连接流。</li></ul>
所有移动设备（Android 和 iOS）	<ul style="list-style-type: none"><li>通过减少罕见崩溃的发生，提高了阶段的稳定性。</li><li>向 <code>BroadcastSession</code> 添加了一种新 <code>sendTimedMetadata</code> 方法，该方法允许通过与当前广播相同的套接字连接发送字符串。此字符串附有计时信息，可以由 IVS 回放器 SDK 接收。</li><li>当参与者离开某阶段时，在 Android 上调用 <code>onParticipantLeft</code> 或在 iOS 上调用 <code>participantDidLeave</code> 之前，参与者的发布状态将更新为未发布。</li></ul>
<a href="#">Android 广播 SDK 1.10.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/android/</a></p>

平台	下载和更改
	<ul style="list-style-type: none"> <li>修复了以下错误：重新连接到某阶段时轮换会导致其他参与者出现裁剪不良。</li> <li>修复了以下问题：无法将 AudioStageStream 设备转换为 AudioDevice。</li> <li>修复了快速切换 background-to-foreground 应用程序导致订阅的视频流被静音的问题。</li> </ul>
<a href="#">iOS 广播 SDK 1.10.0</a>	<p>不分阶段下载：<a href="https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>分阶段下载：<a href="https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast-Stages.xcframework.zip">https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast-Stages.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/ios/</a></p> <ul style="list-style-type: none"> <li>修复了以下问题：在广播会话被电话中断后，查看者可能会丢失音频。不再需要解决方法（在电话中断后重新启动广播会话）。</li> <li>修复了导致无法存在多个阶段且无法全部播放音频的问题。</li> <li>当网络突然丢失时，正在进行的广播现在将立即停止，而不是等待连接超时。</li> </ul>

## 广播开发工具包大小：Android

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64-v8a	1.517 MB	4.761 MB	5.324 MB	15.028 MB
armeabi-v7a	1.340 MB	3.433 MB	4.370 MB	9.489 MB
x86_64	1.653 MB	5.003 MB	5.802 MB	15.837 MB

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
x86	1.662 MB	5.287 MB	5.621 MB	15.964 MB

## 广播开发工具包大小：iOS

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64	1.56 MB	3.84 MB	5.04 MB	10.85 MB

2023 年 6 月 28 日

## 私有通道的查看者会话撤消

IVS 客户现在能够撤消与身份验证令牌关联的查看者会话，以阻止和停止使用该令牌进行播放。有关更多信息，请参阅：

- [设置私有通道](#) — 我们更改了“令牌架构”部分并添加了“撤消查看者会话”。
- [IVS API 参考](#) — 我们添加了两个端点 ( `StartViewerSessionRevocation` 和 `BatchStartViewerSessionRevocation` ) 和两个对象 ( `BatchStartViewerSessionRevocationError` 和 `BatchStartViewerSessionRevocationViewerSession` ) 。

2023 年 6 月 27 日

## Amazon IVS 广播 SDK：iOS 1.9.1

平台	下载和更改
<a href="#">iOS 广播 SDK 1.9.1</a>	不分阶段下载： <a href="https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast.xcframework.zip</a>

平台	下载和更改
	<p>分阶段下载：<a href="https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast-Stages.xcframework.zip">https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast-Stages.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.9.1/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.9.1/ios/</a></p> <ul style="list-style-type: none"> <li>解决了 iOS 16.5 及更高版本上的一个问题，即视频比特率在以下情况下逐渐下降：(1) 如果在关闭 b 帧的情况下不使用自动比特率，则大约 20 分钟；或者 (2) 在到达 <code>IVSVideoConfiguration.maxBitrate</code> 大约 20 分钟后，网络连接保持稳定，关闭 b 帧。</li> </ul> <p>已知问题：在广播会话被电话中断后，观众可能会丢失音频。解决方法是在电话中断后重新启动广播会话。</p>

## 广播开发工具包大小：iOS

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64	1.55MB	3.77MB	5.01 MB	10.77MB

2023 年 6 月 27 日

## Amazon IVS 广播 SDK 1.7.5

平台	下载和更改
<a href="#">iOS 广播 SDK 1.7.5</a>	<p>下载：<a href="https://broadcast.live-video.net/1.7.5/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.7.5/AmazonIVSBroadcast.xcframework.zip</a></p>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.5/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.5/ios/</a></p> <ul style="list-style-type: none"> <li>解决了 iOS 16.5 及更高版本上的一个问题，即视频比特率在以下情况下逐渐下降：(1) 如果在关闭 b 帧的情况下不使用自动比特率，则大约 20 分钟；或者 (2) 在到达 IVSVideoC onfiguration.maxBitrate 大约 20 分钟后，网络连接保持稳定，关闭 b 帧。</li> </ul>

广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.48MB	3.43 MB

2023 年 6 月 16 日

Amazon IVS 广播 SDK：Web 1.3.3

平台	下载和更改
<a href="#">Web 广播 SDK 1.3.3</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"> <li>修复了用于支持 IVS 多主机运行状况的内部分析中的回归问题。</li> </ul>

## 2023 年 6 月 2 日

### 高级通道类型

除了现有的 BASIC 和 STANDARD 类型以外，此版本还引入了两种新的通道类型（ADVANCED\_SD 和 ADVANCED\_HD）。通道类型决定了可接受的播放和录制分辨率和比特率。

- **ADVANCED\_SD**：视频已转码；从原始输入中生成多种品质，以根据观众的设备 and 网络情况自动为他们提供最佳体验。输入分辨率最高可达 1080p，比特率最高可达 8.5 Mbps；输出质量最高为标清（480p）。您可以选择可选的转码预设（见下文）。所有渲染的音频都经过转码，并且还提供了仅限音频的渲染。
- **ADVANCED\_HD**：视频已转码；从原始输入中生成多种品质，以根据观众的设备 and 网络情况自动为他们提供最佳体验。输入分辨率最高可达 1080p，比特率最高可达 8.5 Mbps；输出质量最高为高清（720p）。您可以选择可选的转码预设（见下文）。所有渲染的音频都经过转码，并且还提供了仅限音频的渲染。

新通道类型的可选转码预设允许您在可用的下载带宽和视频质量之间进行权衡，以优化观看体验。预设有两种：

- 受限的带宽传输对每个质量级别使用较低的比特率。如果您的下载带宽较低和/或视频内容简单（例如，讲话者头部特写），则请使用它。
- 较高的带宽传输对每个质量级别使用较高的比特率。如果您有较高的下载带宽和/或复杂的视频内容（例如闪光灯和快速场景切换），则请使用它。

[文档历史记录](#)页面列出了与 IVS 用户指南和 IVS API 参考相关的更改。

## 2023 年 6 月 1 日

### Amazon IVS 广播 SDK：Android 1.9.0 和 iOS 1.9.0

平台	下载和更改
全部	<ul style="list-style-type: none"><li>• 通过减少罕见的崩溃事件提高了阶段的稳定性。</li><li>• 增强了对反复出现的网络中断的自动恢复。</li></ul>

平台	下载和更改
<a href="#">Android 广播 SDK 1.9.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/android/</a></p> <ul style="list-style-type: none"><li>• 修复了一个漏洞，即当用户尝试在某些设备上切换到非默认麦克风时断开有线耳机会导致错误。</li><li>• 修复了某些设备上的漏洞，即在广播会话期间切换麦克风时麦克风连接不正确。</li><li>• 修复了在创建 Broadcast Session、Stage 或 DeviceDiscovery 对象之前调用 Presets 方法时出现的崩溃问题。</li></ul>

平台	下载和更改
<a href="#">iOS 广播 SDK 1.9.0</a>	<p>不分阶段下载：<a href="https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>分阶段下载：<a href="https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast-Stages.xcframework.zip">https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast-Stages.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/ios/</a></p> <ul style="list-style-type: none"><li>• 当您加入舞台但舞台上没有连接麦克风时，进入背景时新的默认行为是停留在舞台中，而不是自动离开。这使得仅限观众的参加者在后台能够收听舞台的情况。</li><li>• 改进了舞台蓝牙设备连接/断开连接的处理。</li><li>• 修复了使用 <code>setGain</code> 静音和取消静音时音频过低的问题。</li><li>• 将相机连接到 <code>IVSBroadcastSession</code> 时，相机现在会根据 <code>size</code> 仅限自我配置，并根据 <code>IVSVideoConfiguration</code> 仅限 <code>targetFramerate</code> 的配置。</li><li>• iOS SDK 现在需要 iOS 12.0 或更高版本（iOS 11 不再受支持。）</li></ul> <p>已知问题：在广播会话被电话中断后，观众可能会丢失音频。解决方法是在电话中断后重新启动广播会话。</p>



## 广播开发工具包大小：Android

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64-v8a	1.638MB	4.846MB	5.451MB	14.778MB
armeabi-v7a	1.461MB	3.532MB	4.506MB	9.475MB
x86_64	1.770MB	5.082MB	5.753MB	15.904MB
x86	1.781MB	5.366MB	5.919MB	15.708MB

## 广播开发工具包大小：iOS

Architectur9	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64	1.55MB	3.77MB	5.00MB	10.77MB

2023 年 5 月 23 日

## Amazon IVS 回放器 SDK 1.19.0

平台	下载和更改
<a href="#">Web 播放器 1.19.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p>NPM Package : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.19.0/amazon-ivs-player.min.js">https://player.live-video.net/1.19.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.19.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.19.0/amazon-ivs-videojs-tech.min.js</a></p>

平台	下载和更改
	参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.19.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.19.0/web/</a>
<a href="#">Android 播放器 1.19.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.19.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.19.0/android/</a> <ul style="list-style-type: none"> <li>修复了自动画质模式中的一个问题，即便有足够的带宽可以切换，播放器在重新缓冲缓冲后仍保持最低画质。</li> </ul>
<a href="#">iOS 播放器 1.19.0</a>	下载： <a href="https://player.live-video.net/1.19.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.19.0/AmazonIVSPlayer.xcframework.zip</a> 参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.19.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.19.0/ios/</a>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.013MB	2.866MB
armeabi-v7a	0.919MB	2.272MB
x86_64	1.084MB	3.001MB
x86	1.058MB	2.702MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB

# 2023 年 5 月 16 日

## Amazon IVS 广播 SDK : iOS 1.8.1

平台	下载和更改
<a href="#">iOS 广播 SDK 1.8.1</a>	<p>不分阶段下载 : <a href="https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast.xcframework.zip</a></p> <p>分阶段下载 : <a href="https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast-Stages.xcframework.zip">https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast-Stages.xcframework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.8.1/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.8.1/ios/</a></p> <ul style="list-style-type: none"> <li>修复了 iOS 16.4 上比特率下降的问题。我，既适用于 RTMP (无舞台)，也适用于 WebRTC (有舞台)。如果您已在应用程序上实施解决方法 (通过启用 b frame)，则可以在安装此更新后将其删除。</li> </ul>

## 广播开发工具包大小 : iOS

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64	1.53MB	3.73MB	5.00MB	10.73MB

# 2023 年 5 月 16 日

## Amazon IVS 广播 SDK 1.7.4

平台	下载和更改
<a href="#">iOS 广播 SDK 1.7.4</a>	<p>下载：<a href="https://broadcast.live-video.net/1.7.4/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.7.4/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.4/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.4/ios/</a></p> <ul style="list-style-type: none"> <li>修复了 iOS 16.4 上比特率下降的问题。如果您已在应用程序上实施解决方法（通过启用 b frame），则可以在安装此更新后将其删除。</li> </ul>

广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.48MB	3.40MB

# 2023 年 5 月 11 日

## 多台主机运行状况

现在，您可以使用多台主机，监控 IVS 舞台的运行状况。请参阅：

- [监控舞台运行状况](#)：这是新的 Amazon IVS 用户指南页面。
- [将亚马逊 EventBridge 与 Amazon IVS 配合使用](#) — 我们添加了两个阶段更新事件。
- [IVS 服务限额](#)：为新端点添加了调用速率限额。
- [IVS Stage API 参考](#) — 我们添加了五个端点（GetParticipant ListParticipants GetStageSession、ListStageSessions、ListParticipantEvents）和五个对象（事件、参与者、ParticipantSummary、StageSession、StageSessionSummary）。

## 2023 年 5 月 1 日

### Amazon IVS Web 广播 SDK 1.3.2

平台	下载和更改
<a href="#">Web 广播 SDK Web 1.3.2</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"><li>• 修复了广播屏幕共享有时会导致直播通道黑屏的问题。</li><li>• 修复了广播舞台参与者有时会导致直播通道黑屏的问题。</li><li>• 修复了报告重复的取消发布和发布事件的分析问题。</li><li>• 修复了 RemoteStageStream 对象并不总是更新的问题。getStats</li><li>• 修复了尝试广播舞台参与者时出现 OverconstrainedError 的问题。</li><li>• 添加了增强功能：当舞台策略 shouldPublishParticipant 设置为 true 时，将忽略仅订阅的参与者。</li></ul>

## 2023 年 4 月 27 日

### 增加舞台参与者

可以同时连接到舞台的最大参与者数量从 12 增加到 1,000。最多可以有 12 名参与者同时发布到舞台，且最多可以有 1,000 人同时订阅舞台。有关更多信息，请参阅[在 Amazon IVS 流中启用多台主机](#)和[Amazon IVS 服务限额](#)。

2023 年 4 月 4 日

## Amazon IVS 回放器 SDK 1.18.0

平台	下载和更改
<a href="#">Web 播放器 1.18.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p>下载和更改</p> <p>NPM Package : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.18.0/amazon-ivs-player.min.js">https://player.live-video.net/1.18.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.18.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.18.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.18.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.18.0/web/</a></p> <ul style="list-style-type: none"> <li>修复了 Safari 问题：刷新后在控制台选项卡中显示“HTTP 响应错误”-“加载失败”。</li> </ul>
<a href="#">Android 播放器 1.18.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.18.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.18.0/android/</a></p> <ul style="list-style-type: none"> <li>修复了当播放速率大于 1x 时视频播放的问题。</li> </ul>
<a href="#">iOS 播放器 1.18.0</a>	<p>下载 : <a href="https://player.live-video.net/1.18.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.18.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.18.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.18.0/ios/</a></p>

移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.011MB	2.854MB

架构	压缩大小	未压缩大小
armeabi-v7a	0.916MB	2.261MB
x86_64	1.082MB	2.990MB
x86	1.055MB	2.691MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB

2023 年 3 月 30 日

## RTMP 支持

除了 RTMPS，Amazon IVS 现在支持 RTMP（实时消息协议）流。RTMPS 是 RTMP 的安全版本。我们建议使用 RTMPS 进行安全提取，除非是在需要 RTMP 的特定且经过验证的使用场景。

可以通过以下方式设置 RTMP 流式传输：

- IVS 控制台：在初始通道设置时使用自定义配置按钮，或者在修改现有通道时使用启用 RTMP 提取开关。
- API — 使用 `CreateChannel` 或 `UpdateChannel` 请求中的新 `insecureIngest` 字段。请参阅 [IVS API 参考](#)。

有关 RTMP 提取端点的信息，请参阅 [设置流式传输软件](#)、[Android 广播 SDK 指南](#) 和 [iOS 广播 SDK 指南](#)。

2023 年 4 月 29 日更正：更改了 [Android 广播 SDK 指南](#) 和 [iOS 广播 SDK 指南](#)，以表明这些 SDK 仅支持 RTMPS 摄取（不支持不安全的 RTMP 摄取）。

## 2023 年 3 月 29 日

### 私人频道的一次性令牌

已在[生成并签署回放令牌](#)中，向负载中添加了可选字段 `single-use-uuid`，用于生成一次性令牌。

## 2023 年 3 月 28 日

### Amazon IVS Web 广播 SDK 1.3.1

平台	下载和更改
<a href="#">Web 广播 SDK : Web 1.3.1</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a> <ul style="list-style-type: none"><li>此版本中没有面向客户的更改。</li></ul>

## 2023 年 3 月 23 日

### 在流上支持多台主机（舞台资源）

这是新功能的第一个版本：您现在可以将来自多个参与者的视频合并为一个实时流。舞台是一个虚拟空间，参与者可以在其中实时交流音频和视频。然后您可以将“舞台”广播到通道让更多的观众看到，并构建让观众“登上舞台”参与实时对话的应用程序。有关详细信息，请参阅：

- [在 IVS 流中启用多台主机](#)（新文档）
- [舞台 API 参考](#)（新文档）
- [服务限额](#)（参见其他限额 > Amazon IVS 中的“Amazon IVS 舞台”端点和舞台限制）
- 同步发布 [Amazon IVS 广播 SDK 1.8.0](#) 的文档变更

## 2023 年 3 月 23 日

### Amazon IVS 广播 SDK : Android 1.8.0、iOS 1.8.0 和 Web 1.3.0

除了增加在流上支持多台主机外，还更新了 Android 和 iOS 广播 SDK 以支持新的舞台功能。



平台	下载和更改
全部	<ul style="list-style-type: none"> <li>• 添加了舞台支持。请参阅<a href="#">在流上支持多台主机 (舞台资源)</a>。</li> <li>• 已在<a href="#">广播 SDK 概述</a>中添加舞台平台要求。</li> </ul>
<a href="#">Android 广播 SDK 1.8.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/android/</a></p> <ul style="list-style-type: none"> <li>• 根据您的 Gradle 文件的配置方式（使用分类器），您可以下载支持或不支持舞台的 SDK。不支持舞台的 SDK 较小。有关详细信息，请参阅<a href="#">广播 SDK：Android 指南</a>。</li> <li>• 在<a href="#">广播 SDK：Android 指南</a>中，添加了“添加多台带有舞台功能 SDK 的主机”和与舞台相关的“已知问题和解决方法”。</li> </ul>
<a href="#">iOS 广播 SDK 1.8.0</a>	<p>下载不带舞台功能的 SDK：<a href="https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>下载带舞台功能的 SDK：<a href="https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast-Stages.xcframework.zip">https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast-Stages.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/ios/</a></p> <ul style="list-style-type: none"> <li>• 根据您的 pod 文件的配置方式，您可以下载支持或不支持舞台的 SDK。不支持舞台的 SDK 较小。有关安装的详细信息，请参阅<a href="#">广播 SDK：iOS 指南</a>。</li> <li>• 在<a href="#">广播 SDK：iOS 指南</a>中，添加了“添加多台带有舞台功能 SDK 的主机”和与舞台相关的“已知问题和解决方法”。</li> <li>• 已弃用 SDK 的位码支持，因为 Apple 已经正式弃用了位码，应用商店提交项不再接受位</li> </ul>

平台	下载和更改
	码。有关更多信息，请参阅 <a href="#">Xcode 14 发布说明</a> 。
<a href="#">Web 广播 SDK : Web 1.3.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"> <li>在<a href="#">广播 SDK : Web 指南</a>中，添加了“添加多台带有舞台功能 SDK 的主机”和与舞台相关的“已知问题和解决方法”。</li> </ul>

### 广播开发工具包大小：Android

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64-v8a	1.767MB	5.192MB	5.886MB	16.398MB
armeabi-v7a	1.656MB	4.263MB	4.946MB	10.924MB
x86_64	1.967MB	5.735MB	6.316MB	17.376 MB
x86	1.894MB	5.196MB	6.387MB	16.730MB

### 广播开发工具包大小：iOS

架构	不带舞台功能的压缩大小	不带舞台功能的未压缩大小	带有舞台功能的压缩大小	带有舞台功能的未压缩大小
arm64	1.53MB	3.73MB	5.03MB	10.67MB

## 2023 年 3 月 2 日

### Amazon IVS 广播 SDK : Android 1.7.3

平台	下载和更改
<a href="#">Android 广播 SDK 1.7.3</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.3/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.3/android/</a> <ul style="list-style-type: none"> <li>修复了自定义图像源无法在搭载 D MediaTek imensity 700 SoC 的设备上正常运行的问题。</li> </ul>

### 广播开发工具包大小 : Android

架构	压缩大小	未压缩大小
arm64-v8a	1.629MB	4.688MB
armeabi-v7a	1.520MB	3.792MB
x86_64	1.825MB	5.218MB
x86	1.629MB	4.688MB

## 2023 年 2 月 28 日

### Amazon IVS 回放器 SDK 1.17.0

平台	下载和更改
<a href="#">Web 播放器 1.17.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	NPM Package： <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a> 脚本资产： <a href="https://player.live-video.net/1.17.0/amazon-ivs-player.min.js">https://player.live-video.net/1.17.0/amazon-ivs-player.min.js</a>

平台	下载和更改
	<p>Video.js 科技资产：<a href="https://player.live-video.net/1.17.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.17.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.17.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.17.0/web/</a></p> <ul style="list-style-type: none"> <li>实现了对移动版 Safari getLiveLatency 方法的支持。</li> </ul>
<a href="#">Android 播放器 1.17.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.17.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.17.0/android/</a></p>
<a href="#">iOS 播放器 1.17.0</a>	<p>下载：<a href="https://player.live-video.net/1.17.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.17.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.17.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.17.0/ios/</a></p> <ul style="list-style-type: none"> <li>已弃用 SDK 的位码支持，因为 Apple 已经正式弃用了位码，应用商店提交项不再接受位码。有关更多信息，请参阅 <a href="#">Xcode 14 发行说明</a>。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.009MB	2.853MB
armeabi-v7a	0.915MB	2.260MB
x86_64	1.081MB	2.988MB
x86	1.054MB	2.690MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB

2023 年 2 月 16 日

## 用于自动录制到 S3 的字节范围标签和清单文件

除了标准的 auto-record-to HLS 播放列表外，-S3 功能现在还支持生成[字节范围的播放列表](#)。有关更多信息，请参阅[自动录制到 Amazon S3](#)（ recording\_started 和 recording\_ended 的 JSON 示例中的“录制内容”、“字节范围播放列表”和新的 byte\_range\_playlist 字段。）

2023 年 1 月 31 日

## Amazon IVS Chat 客户端消息收发 SDK：Android 1.1.0

平台	下载和更改
<a href="#">Android 聊天功能客户端消息收发 SDK 1.1.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.1.0/">https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.1.0/</a></p> <ul style="list-style-type: none"> <li>为了支持 Kotlin 协同例程，我们在 com.amazonaws.ivs.chat.messaging.coroutines 程序包中添加了新的 IVS 聊天功能消息收发 API。另请参阅新的 Kotlin 协同教程；第 1 部分（共 2 部分）是<a href="#">聊天室</a>。</li> </ul>

## Chat 客户端消息收发 SDK 大小：Android

架构	压缩大小	未压缩大小
所有架构（字节码）	89KB	92KB

2023 年 1 月 17 日

## Amazon IVS 回放器 SDK 1.16.0

平台	下载和更改
<p><a href="#">Web 回放器 1.16.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 回放器集成</a></p>	<p><b>下载和更改</b></p> <p><b>NPM Package</b> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产</b> : <a href="https://player.live-video.net/1.16.0/amazon-ivs-player.min.js">https://player.live-video.net/1.16.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产</b> : <a href="https://player.live-video.net/1.16.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.16.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.16.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.16.0/web/</a></p> <ul style="list-style-type: none"> <li>更新了 SDK 文档，以说明 iOS 移动浏览器不支持的方法。</li> </ul>
<p><a href="#">Android 回放器 1.16.0</a></p>	<p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.16.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.16.0/android/</a></p> <ul style="list-style-type: none"> <li>增加了 setOrigin 方法以允许在回放请求中包含 Origin 请求标头。另请参阅 <a href="#">令牌 Schema</a>，以了解有关新的 strict-origin-enforcement 字段的信息。</li> </ul>
<p><a href="#">iOS 回放器 1.16.0</a></p>	<p><b>下载</b> : <a href="https://player.live-video.net/1.16.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.16.0/AmazonIVSPlayer.xcframework.zip</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.16.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.16.0/ios/</a></p> <ul style="list-style-type: none"> <li>增加了 setOrigin 方法以允许在回放请求中包含 Origin 请求标头。另请参阅 <a href="#">令牌 Schema</a>，以了解有关新的 strict-origin-enforcement 字段的信息。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.009 MB	2.852 MB
armeabi-v7a	0.914 MB	2.258 MB
x86_64	1.054 MB	2.689 MB
x86	1.080 MB	2.987 MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB

2022 年 12 月 9 日

## 向自动记录到 S3 清单文件添加了时间戳

启用 [Auto-Record to Amazon S3](#) ( 自动录制到 Amazon S3 ) 后，将创建 HLS 清单文件。这些文件现在包含 HLS Program-Date-Time (PDT) 标签，PDT 标签使用 UTC ISO-8601 格式表示每个 HLS 片段在生成时的挂钟时间。

2022 年 12 月 6 日

## Amazon IVS 广播 SDK：Android 1.7.2

平台	下载和更改
<a href="#">Android 广播 SDK 1.7.2</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.2/an-droid/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.2/an-droid/</a>

平台	下载和更改
	<ul style="list-style-type: none"> <li>修复了以下错误：非摄像头设备子类 <code>SurfaceSource</code> 返回的 <code>Device.Descriptor</code> 将在每次调用时提供唯一的 <code>deviceId</code> 和 <code>urn</code>，导致无法通过这些属性可靠地识别设备。</li> <li>修复了以下错误：如果在添加相关的插槽时其 <code>preferredAudioInput</code> 值为 <code>Device.Descriptor.DeviceType.MICROPHONE</code>，则当 <code>Mixer.getSlots()</code> 进行查询时 <code>BroadcastConfiguration.Mixer.Slot</code> 的 <code>preferredAudioInput</code> 属性为空。</li> </ul>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.628MB	4.682MB
armeabi-v7a	1.519MB	3.786MB
x86_64	1.701 MB	5.075 MB
x86	1.637 MB	4.605 MB

## 2022 年 11 月 17 日

### 聊天记录

这是新功能的第一个版本。现在，您可以创建日志记录配置来存储发送到聊天室的消息。有关更多信息，请参阅：

- [聊天记录](#) – 新页面。
- [Getting Started with Chat](#) – 更新了 IAM 权限，添加了设置聊天记录的程序。



- [Service Quotas](#) – 新端点和日志记录配置。
- CloudWatch — 添加了日志目标指标。

2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring [IVS 低延迟流媒体](#)。

2023 年 12 月 28 日更新：聊天相关 CloudWatch 内容已移至“监控[亚马逊 IVS 聊天](#)”。

- [Chat API 参考](#)-添加了一个 LoggingConfiguration 资源以及几种数据类型和端点。有关详细信息，请参阅[文档历史记录](#)。

## 2022 年 11 月 9 日

### 亚马逊 IVS 聊天客户端消息传送软件开发工具包：1.0.2 JavaScript

平台	下载和更改
<a href="#">JavaScript 聊天客户端消息 SDK 1.0.2</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.2/">https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.2/</a> <ul style="list-style-type: none"> <li>• 修复了影响 Firefox 的问题：客户端在使用端点与聊天室断开连接时错误地收到套接字错误。DisconnectUser</li> </ul>

## 2022 年 11 月 1 日

### Amazon IVS 播放器 SDK 1.14.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>• 通过减少罕见的崩溃事件提高了播放器的稳定性。</li> </ul>
<a href="#">Web 播放器 1.14.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	NPM Package： <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a>  脚本资产： <a href="https://player.live-video.net/1.14.0/amazon-ivs-player.min.js">https://player.live-video.net/1.14.0/amazon-ivs-player.min.js</a>

平台	下载和更改
	<p>Video.js 科技资产 : <a href="https://player.live-video.net/1.14.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.14.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.14.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.14.0/web/</a></p>
<a href="#">Android 播放器 1.14.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.14.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.14.0/android/</a></p> <ul style="list-style-type: none"> <li>向 <code>ErrorType</code> 类添加了 <code>getErrorCode()</code> 方法。</li> </ul>
<a href="#">iOS 播放器 1.14.0</a>	<p>下载 : <a href="https://player.live-video.net/1.14.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.14.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.14.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.14.0/ios/</a></p> <ul style="list-style-type: none"> <li>公布 IVS 播放器 <code>setQuality:adaptive:</code> 方法。</li> </ul>

## 移动开发工具包大小 : Android

架构	压缩大小	未压缩大小
arm64-v8a	1.004 MB	2.840 MB
armeabi-v7a	0.909 MB	2.248 MB
x86_64	1.049 MB	2.678 MB
x86	1.075 MB	2.975 MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.41 MB	0.99 MB

2022 年 10 月 18 日

## 亚马逊 IVS 聊天客户端消息传送软件开发工具包：1.0.1 JavaScript

平台	下载和更改
<a href="#">JavaScript 聊天客户端消息 SDK 1.0.1</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.1/">https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.1/</a>

2022 年 10 月 6 日

## Amazon IVS 广播 SDK 1.7.1

平台	下载和更改
<a href="#">iOS 广播 SDK 1.7.1</a>	<p>下载：<a href="https://broadcast.live-video.net/1.7.1/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.7.1/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.1/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.1/ios/</a></p> <ul style="list-style-type: none"> <li>修复了直接链接到几个类时出现的链接器错误。</li> <li>删除了不应由主机应用程序实例化的类上的 <code>init</code> 和 <code>new</code> 函数。</li> <li>使用 SDK 提供的摄像头并配置为 9:16 竖屏宽高比的插槽现在可以正确使用匹配的 9:16 摄像头宽高比。（这些插槽以前使用的是 3:4 摄</li> </ul>

平台	下载和更改
	像头宽高比。) 使用 FIT 宽高比模式的插槽现在使用全屏。(这些插槽以前使用的是宽银幕。)

广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.48MB	3.40MB

2022 年 9 月 22 日

## Amazon IVS 广播 SDK 1.7.0

注意：尚未推出 1.6.0 版本。

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>通过减少罕见的崩溃事件提高了稳定性。</li> <li>在上面添加了一个AutomaticBitrateProfile 枚举。VideoConfiguration此枚举可控制 ABR 算法调整视频比特率的速率。</li> <li>增加了 onTransmissionStatsChanged 方法。它包含比 onBroadcastQualityChanged 和 onNetworkHealthChanged 更详细的传输统计数据。我们已弃用后两种方法，建议您改用 onTransmissionStatsChanged。</li> </ul>
<a href="#">Android 广播 SDK 1.7.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/android/</a>
<a href="#">iOS 广播 SDK 1.7.0</a>	下载： <a href="https://broadcast.live-video.net/1.7.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.7.0/AmazonIVSBroadcast.xcframework.zip</a>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/ios/</a></p> <ul style="list-style-type: none"> <li>增加了 <code>IVSBroadcastSessionAudioSessionStrategy.PlayAndRecordDefaultToSpeaker</code>，让开发人员能够指定带有听筒的设备（例如 iPhone）是否优先使用扬声器播放，而不是耳机。</li> </ul>

### 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.628MB	4.682MB
armeabi-v7a	1.519MB	3.786MB
x86_64	1.824MB	5.212MB
x86	1.760MB	4.742MB

### 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.47MB	3.40MB

# 2022 年 9 月 20 日

## Amazon IVS 播放器 SDK 1.13.0

平台	下载和更改
<a href="#">Web 播放器 1.13.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><b>下载和更改</b></p> <p><b>NPM Package :</b> <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产 :</b> <a href="https://player.live-video.net/1.13.0/amazon-ivs-player.min.js">https://player.live-video.net/1.13.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产 :</b> <a href="https://player.live-video.net/1.13.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.13.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档 :</b> <a href="https://aws.github.io/amazon-ivs-player-docs/1.13.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.13.0/web/</a></p> <ul style="list-style-type: none"><li>• 增加了对 VideoJS seeking() 函数的支持。</li><li>• 删除了导致开发问题的未使用类型 (CaptureEventTypes)。</li><li>• 修复了网络恢复时 MediaSource 出现的间歇性错误。</li></ul> <p><b>已知问题 :</b> 打开控制台时, 可能会出现“Sawmill Enabled” (Sawmill 已启用) 日志。此内部日志应处于隐藏状态, 因为它不会对客户产生影响。如果看到该日志, 忽略即可。</p>
<a href="#">Android 播放器 1.13.0</a>	<p><b>参考文档 :</b> <a href="https://aws.github.io/amazon-ivs-player-docs/1.13.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.13.0/android/</a></p> <ul style="list-style-type: none"><li>• 增加了额外的防护机制, 以防止与竞争条件有关的播放崩溃。</li><li>• 对 ABR 带宽估算执行稳定性改进。</li></ul>

平台	下载和更改
<a href="#">iOS 播放器 1.13.0</a>	<p>下载：<a href="https://player.live-video.net/1.13.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.13.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.13.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.13.0/ios/</a></p> <ul style="list-style-type: none"> <li>• 修复了在后台播放时纯音频播放可能会停止这一错误。</li> <li>• 增加了额外的防护机制，以防止与竞争条件有关的播放崩溃。</li> <li>• 对 ABR 带宽估算执行稳定性改进。</li> <li>• 在 SDK 参考中阐明了 <code>setAutoMaximumQuality</code> 根据比特率筛选质量。</li> <li>• 更改了 <code>IVSPlayer</code> 类的 <code>setQuality:</code> 方法，使其忽略无效值。</li> </ul>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.000MB	2.829MB
armeabi-v7a	0.904MB	2.237 MB
x86_64	1.070MB	2.962MB
x86	1.045MB	2.665 MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.44MB	1.06MB

## 2022 年 9 月 15 日

### 竖屏视频改进 ( 最终版本 )

今天，我们开始为所有 Amazon IVS 客户推出[竖屏视频改进](#)中记录的更改。将更改应用于所有账户将需要 2 到 3 天。

## 2022 年 9 月 12 日

### Amazon IVS 广播 SDK 1.5.2 : iOS

平台	下载和更改
<a href="#">iOS 广播 SDK 1.5.2</a>	<p>下载：<a href="https://broadcast.live-video.net/1.5.2/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.5.2/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.5.2/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.5.2/ios/</a></p> <ul style="list-style-type: none"><li>修复了以下罕见崩溃问题：在广播停止后但完成关闭广播之前，网络连接很快就中断。</li><li>修复了重试循环在发生致命错误后反复尝试重新启动广播时出现的内存增长问题。</li></ul>

### 广播开发工具包大小 : iOS

架构	压缩大小	未压缩大小
arm64	1.39MB	3.20 MB



2022 年 9 月 8 日

## Amazon IVS Chat 客户端消息收发 SDK : Android 1.0.0 和 iOS 1.0.0

平台	下载和更改
<a href="#">Android Chat 客户端消息收发 SDK 1.0.0</a>	参考文档 : <a href="https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.0.0/">https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.0.0/</a>
<a href="#">iOS Chat 客户端消息收发 SDK 1.0.0</a>	参考文档 : <a href="https://aws.github.io/amazon-ivs-chat-messaging-sdk-ios/1.0.0/">https://aws.github.io/amazon-ivs-chat-messaging-sdk-ios/1.0.0/</a>

### Chat 客户端消息收发 SDK 大小 : Android

架构	压缩大小	未压缩大小
所有架构 ( 字节码 )	53KB	58KB

### Chat 客户端消息收发 SDK 大小 : iOS

架构	压缩大小	未压缩大小
ios-arm64_x86_64 模拟器 ( 位码 )	484KB	2.4MB
ios-arm64_x86_64 模拟器	484KB	2.4MB
ios-arm64 ( 位码 )	1.1MB	3.1MB
ios-arm64	233KB	1.2MB

## 2022 年 9 月 2 日

### Amazon IVS Web 广播 SDK 1.2.0

平台	下载和更改
<a href="#">Web 广播 SDK</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"><li>• 修复了运行 npm 包时的捆绑包类型定义。</li><li>• 添加了支持 1080 BASIC 通道的预设配置。</li></ul>

## 2022 年 8 月 30 日

### 合并片段化的流

这是新功能的第一个版本。如果将流配置为自动录制到 Amazon S3，则您现在可以指定一个时段，在此期间，如果流中断并启动了新流，Amazon IVS 会尝试录制到与上一个流相同的 S3 前缀。换句话说，如果广播断开连接，然后在指定的间隔内重新连接，则多个流将被视为一个广播并合并在一起。有关更多信息，请参阅：

- Amazon IVS 入门 - 我们更新了[第 3 步：使用可选录制创建通道](#)，以获得控制台和 CLI 说明。
- 自动录制到 S3 - 请参阅新部分[合并片段化的流](#)。
- EventBridge — 在“[示例：记录状态更改](#)”中 recording\_session\_id，添加了 recording\_session\_stream\_ids 字段。
- [IVS API 参考](#) — 我们在 CreateRecordingConfiguration 请求和 RecordingConfiguration 对象中添加了 recordingReconnectWindowSeconds 字段。这会影三个响应（CreateRecordingConfiguration GetRecordingConfiguration、和 GetStreamSession）。

## 2022 年 8 月 9 日

### Amazon IVS Web 播放器 SDK 1.12.0

平台	下载和更改
<a href="#">Web 播放器 1.12.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><b>NPM Package :</b> <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产 :</b> <a href="https://player.live-video.net/1.12.0/amazon-ivs-player.min.js">https://player.live-video.net/1.12.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产 :</b> <a href="https://player.live-video.net/1.12.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.12.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档 :</b> <a href="https://aws.github.io/amazon-ivs-player-docs/1.12.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.12.0/web/</a></p> <ul style="list-style-type: none"> <li>增加了额外的防护机制，以防止与竞争条件有关的播放崩溃。</li> </ul>

## 2022 年 7 月 28 日

### Amazon IVS iOS 广播 SDK 1.5.1

平台	下载和更改
<a href="#">iOS 广播 SDK 1.5.1</a>	<p><b>下载 :</b> <a href="https://broadcast.live-video.net/1.5.1/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.5.1/AmazonIVSBroadcast.xcframework.zip</a></p> <p><b>参考文档 :</b> <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.5.1/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.5.1/ios/</a></p> <ul style="list-style-type: none"> <li>修复了与 iOS 16 的兼容性问题，该问题会阻止音频编码，从而导致所有广播失败。此问题会影响适用于 iOS 的 IVS 广播 SDK 的所有先</li> </ul>

平台	下载和更改
	<p>前版本。在 iOS 16 上进行广播必须使用版本 1.5.1。</p> <ul style="list-style-type: none"> <li>修复了直接向 <code>IVSBroadcastSession</code> 的初始化程序提供委托时的内存泄露问题。（一种解决方法是在之后设置委托属性。）</li> </ul>

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.36 MB	3.20 MB

2022 年 7 月 21 日

## Amazon IVS Web 广播 SDK

平台	下载和更改
<a href="#">Web 广播 SDK</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference">https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</a></p> <ul style="list-style-type: none"> <li>这是 Amazon IVS Web 广播 SDK 的首次发布。</li> </ul>

2022 年 7 月 14 日

## Amazon IVS iOS 播放器 SDK 1.8.3

平台	下载和更改
<a href="#">iOS 播放器 1.8.3</a>	<p>下载：<a href="https://player.live-video.net/1.8.3/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.8.3/AmazonIVSPlayer.xcframework.zip</a></p>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.3/ios/">https://aws.github.io/amazon-ivs-player-docs/1.8.3/ios/</a></p> <ul style="list-style-type: none"> <li>修复了播放器无法播放通过 URL ( 其中包含相对路径 ) 提供的录制内容的问题。</li> <li>修复了主线程被阻止时可能出现的内存增大问题。</li> </ul>

移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.46 MB	1.10 MB

2022 年 6 月 28 日

Amazon IVS 播放器 Web SDK 1.11.0

平台	下载和更改
<p><a href="#">Web 播放器 1.11.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a></p>	<p>NPM Package：<a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产：<a href="https://player.live-video.net/1.11.0/amazon-ivs-player.min.js">https://player.live-video.net/1.11.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产：<a href="https://player.live-video.net/1.11.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.11.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.11.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.11.0/web/</a></p> <ul style="list-style-type: none"> <li>对 ABR 带宽估算执行稳定性改进。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>修复使用 Video.js 集成在 iOS 移动浏览器上播放录制内容时出现的问题：重新播放按钮现在可正常工作。不再需要之前的解决办法（初始化 Video.js 时隐藏重新播放按钮）。</li> </ul>

2022 年 6 月 22 日

## Amazon IVS 广播 SDK 1.5.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>通过减少罕见的崩溃事件提高了稳定性。</li> <li>提高了高比特率流的稳定性。</li> <li>延迟极高的广播将以错误代码 20401 结束，并显示以下消息：“因为网络延迟太高，广播已经结束。请检查您的连接是否稳定，或者降低广播比特率。”此操作的延迟阈值可能会随着时间的推移而改变；目前为 45 秒。</li> </ul>
<a href="#">Android 广播 SDK 1.5.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/android/</a></p> <ul style="list-style-type: none"> <li>为 Video 添加新的配置选项，以启用广播会话的透明度：<code>enableTransparency</code> (boolean) 和 <code>isTransparencyEnabled()</code>。默认情况下，透明度处于禁用状态。请注意，对于个别插槽的 <code>fillColor</code> 或 <code>transparency</code> 值，您必须将 <code>Video.enableTransparency</code> 设置为 TRUE，方可按预期工作。仅在需要时启用透明度，因为它的计算密集程度更高。</li> </ul>
<a href="#">iOS 广播 SDK 1.5.0</a>	<p>下载：<a href="https://broadcast.live-video.net/1.5.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.5.0/AmazonIVSBroadcast.xcframework.zip</a></p>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/ios/</a></p> <ul style="list-style-type: none"> <li>使用 <code>IVSReplayKitBroadcastSession</code> 进行屏幕共享时，建议您在 <code>RPBroadcastSampleHandler::broadcastFinished</code> 中调用 <code>IVSReplayKitBroadcastSession::broadcastFinished</code> 以确保正确关闭流。如果不这样做，可能会导致流在超时之前一直处于活动状态。</li> <li><code>IVSImagePreviewView</code> 不再受 <code>MTKView</code> 支持，而是改为支持普通的 <code>UIView</code>，其有基于 <code>AVSampleBufferDisplayLayer</code> 的 <code>CALayer</code>。</li> </ul>

### 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.539 MB	4.355 MB
armeabi-v7a	1.431 MB	3.483 MB
x86_64	1.729 MB	4.868 MB
x86	1.675 MB	4.436 MB

### 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.36 MB	3.20 MB

# 2022 年 6 月 9 日

## 垂直视频改进

此版本改进了 Amazon IVS 处理垂直输入的方式；例如，在高度大于宽度的移动设备上广播的内容。正如本文末尾所述，随着时间的推移，这一更改会陆续进行。

共有三大更改：

1. (仅影响 Standard 通道) 垂直输入会根据内容的宽度进行缩放，从而减少缩放并提供更高质量的视觉输出。例如，以下是此更改对 720x1280 输入产生的影响：

名称	旧的宽度 x 高度	新的宽度 x 高度
1280p	720 x 1280	:
720p	404 x 720	720 x 1280
480p	268 x 480	480 x 852
360p	200 x 360	360 x 640
160p	88 x 160	160 x 284

2. (仅影响 Standard 通道) 生成的唯一呈现是宽度小于或等于输入宽度的呈现。例如，如果输入宽度为 720x1280，则会生成 720p、480p、360p 和 160p 呈现。如果输入宽度介于呈现之间，则所有生成的呈现的宽度都会低于输入宽度。例如，以下是此更改对 540x960 输入产生的影响：

名称	旧的宽度 x 高度	新的宽度 x 高度
960p	540 x 960	:
720p	404 x 720	:
480p	268 x 480	480 x 852
360p	200 x 360	360 x 640
160p	88 x 160	160 x 284



3. (会影响 Standard 和 Basic 通道) 垂直输入的呈现会使用更为传统的命名方案，基于宽度而不是高度。例如，Basic 通道的 360x640 输入具有一个名为 360p 的输出呈现。

此名称会作为 NAME 属性，在视频播放列表以及面向用户的质量选择器中 ( [示例](#) ) 显示。该名称还会用作录制资产的 Amazon S3 目录名称。例如，对于 360x640 输入，质量选择器和自动录制到 Amazon S3 目录名称为 360p60 (旧值为 640p60)。

随着时间的推移，这项改进会陆续推进：

- 现在 – 在过去的六个月里，您是否广播过垂直输入？如果没有，我们现在将为您账户启用此更改 (具体来说，是从今天开始的 1 周内)。如果是，您将在 AWS Health Dashboard 的账户事件部分收到有关此更改的通知。
- 2022 年 9 月 15 日 – 我们将对所有剩余账户启用此更改。如果您在过去六个月内广播过垂直输入，并希望尽快在账户上启用此更改，请提交 Amazon Web Services Support 票证。

**重要提示：** 确保您没有任何依赖于旧行为的代码 (例如，录音文件后处理)。例如，如果您的脚本具有硬编码格式的呈现宽度/高度，则必须对其进行编辑，否则可能会在应用此更改后中断。

## 2022 年 5 月 24 日

### Amazon IVS Web 和 Android 播放器开发工具包 1.10.0

平台	下载和更改
<p><a href="#">Web 播放器 1.10.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a></p>	<p><b>NPM Package :</b> <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产 :</b> <a href="https://player.live-video.net/1.10.0/amazon-ivs-player.min.js">https://player.live-video.net/1.10.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产 :</b> <a href="https://player.live-video.net/1.10.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.10.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档 :</b> <a href="https://aws.github.io/amazon-ivs-player-docs/1.10.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.10.0/web/</a></p> <ul style="list-style-type: none"> <li>• 修复了与通过 Video.js 插件进行暂停和播放相关的控制台错误。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>在参考文档中，从 TypeScript 定义文件中删除了两种本来不应该公开的类型，AutoplayOptions 和 PlayerEventType.STATE_CHANGED。</li> <li>修复了使用 setAutoMaxQuality 和 setAutoMaxVideoSize 期间未考虑所有质量的问题。</li> <li>公开了 setAutoMaxVideoSize 方法，并附有相应的文档。</li> <li>在 SDK 参考中阐明了 setAutoMaxQuality 根据比特率筛选质量。</li> <li>更正了网络平台上 VOD 的 end-of-stream 行为。</li> </ul>
<a href="#">Android 播放器 1.10.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.10.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.10.0/android/</a></p> <ul style="list-style-type: none"> <li>修复了使用 setAutoMaxQuality 和 setAutoMaxVideoSize 期间未考虑所有质量的问题。</li> <li>在 Player 类中添加了 getVolume()。</li> <li>在 SDK 参考中阐明了 setAutoMaxQuality 根据比特率筛选质量。</li> <li>更正了网络平台上 VOD 的 end-of-stream 行为。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	0.990 MB	2.805 MB
armeabi-v7a	0.895MB	2.215 MB

架构	压缩大小	未压缩大小
x86_64	1.033 MB	2.643 MB
x86	1.058 MB	2.936 MB

## 2022 年 4 月 28 日

### 流运行状况更新

Amazon IVS 流运行状况有两处更新，可近乎实时地监控实时流的运行状况：

- 高分辨率 CloudWatch 指标图表现已在控制台的直播会话详细信息页面中提供。
- 新维度 (Health) 已添加到 ConcurrentStreams 指标，以按通道运行状况来筛选结果。

参见监控 Amazon IVS Live Stream Health 和使用亚马逊监控亚马逊 IVS。CloudWatch

2023 年 10 月 12 日更新：这些文档已合并到[监控 IVS 低延迟直播功能](#)。

## 2022 年 4 月 26 日

### Amazon IVS Chat

这是 Amazon IVS Chat 的初始版本，Amazon IVS Chat 是一项托管的实时聊天功能，可与实时视频流一起使用。新文档可从[Amazon IVS 文档登陆页面](#)访问。

- 从[Amazon IVS Chat 入门](#)开始。
- 在 Amazon IVS Chat User Guide 中：
  - 请参阅[聊天消息审核处理程序](#)，这是一个新页面。
  - 在使用亚马逊监控亚马逊 IVS CloudWatch、Amazon IVS Security 和 Amazon IVS Service Quotas 中搜索“聊天”更改。

2023 年 10 月 12 日更新：CloudWatch 文档已删除，内容已移至 Monitoring [IVS 低延迟流媒体](#)。

2023 年 12 月 28 日更新：所有聊天信息均收集在新的[Chat User Guide](#)中。

- 文档登陆页面的《Amazon IVS Chat》新增章节包含两个 API 参考：

- [Chat API 参考](#) – 控制面板 API ( HTTPS ) 。
- [聊天消息API参考](#) —数据平面API (WebSocket).

与往常一样，有关文档更改的说明，请参阅 Amazon IVS [文档历史记录](#)。

## 2022 年 4 月 22 日

### Amazon IVS iOS 播放器 SDK 1.8.2

平台	下载和更改
<a href="#">iOS 播放器 1.8.2</a>	<p>下载：<a href="https://player.live-video.net/1.8.2/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.8.2/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.2/ios/">https://aws.github.io/amazon-ivs-player-docs/1.8.2/ios/</a></p> <ul style="list-style-type: none"> <li>• 在运行 iOS 15 及更高版本的设备上增加了对画中画的支持。您可以使用的实例直接实例化 <a href="#">AV PictureInPictureController</a> 类。IVSPlayerLayer 有关示例实施，请参阅<a href="#">公开示例应用程序</a>。</li> <li>• 修复了从 <code>-seekTo:completionHandler:</code> 方法的完成处理程序操作 IVSPlayer 状态时可能出现的锁死问题。</li> <li>• 修复了 1.8.1 发行版在尝试解决主线程被阻止时可能出现的内存增大问题而引入的问题。</li> </ul>

移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.46 MB	1.10 MB

# 2022 年 4 月 19 日

## Amazon IVS 广播 SDK 1.4.0

平台	下载和更改
全部	<ul style="list-style-type: none"><li>通过减少罕见的崩溃事件提高了稳定性。</li><li>增加了一个有关 <a href="#">广播 SDK : 自定义图像源</a> 的新页面。</li></ul>
<a href="#">Android 广播 SDK 1.4.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/android/</a></p> <ul style="list-style-type: none"><li>修复了 <code>createServiceNotificationBuilder</code> 中的一个错误以支持 Android 12。</li><li>通过回退到基准 AVC 配置文件, 修复了主 AVC 配置文件错误的设备上的问题。</li><li>为多个公共 API 方法签名添加一些 <code>NonNull</code> 注解, 以防止意外异常导致应用程序崩溃。</li></ul>
<a href="#">iOS 广播 SDK 1.4.0</a>	<p>下载 : <a href="https://broadcast.live-video.net/1.4.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.4.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/ios/</a></p> <ul style="list-style-type: none"><li>通过更好地利用 GCD 和 Darwin 优化锁以及改善缓冲区的重复使用, 提高了整个 SDK 在 iOS 上的性能。</li><li>在中 <code>BroadcastConfiguration</code>, 将关键帧间隔最大值从 10 更改为 5, 以与 Android 保持一致。</li><li>增加了一种控制音频编码器质量的新方法。在 <code>IVSAudioConfiguration</code> 上, 使用</li></ul>

平台	下载和更改
	<p><code>setQuality</code> 方法。编码器质量下降可能会对 CPU 利用率产生显著影响。</p>

### 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.526 MB	4.324 MB
armeabi-v7a	1.416 MB	3.442 MB
x86_64	1.657 MB	4.393 MB
x86	1.712 MB	4.827 MB

### 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.33MB	3.13 MB

2022 年 3 月 31 日

### Amazon IVS iOS 播放器 SDK 1.8.1

平台	下载和更改
<a href="#">iOS 播放器 1.8.1</a>	<p>下载：&lt;已弃用&gt;</p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.1/ios/">https://aws.github.io/amazon-ivs-player-docs/1.8.1/ios/</a></p> <ul style="list-style-type: none"> <li>在运行 iOS 15 及更高版本的设备上增加了对画中画的支持。您可以使用的实例直</li> </ul>

平台	下载和更改
	<p>接实例化 <a href="#">AV PictureInPictureController</a> 类。IVSPlayerLayer 有关示例实施，请参阅 <a href="#">公开示例应用程序</a>。</p> <ul style="list-style-type: none"> <li>修复了主线程被阻止时可能出现的内存增大问题。</li> <li>修复了从 <code>-seekTo:completionHandler:</code> 方法的完成处理程序操作 IVSPlayer 状态时可能出现的锁死问题。</li> </ul>

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.46 MB	1.10 MB

2022 年 3 月 3 日

### Amazon IVS 广播 SDK 1.3.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>通过减少罕见的崩溃事件提高了稳定性。</li> <li>增加了对 32 位带符号整数和 64 位浮点型 PCM 音频的支持。</li> </ul>
<a href="#">Android 广播 SDK 1.3.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/android/</a></p> <ul style="list-style-type: none"> <li>修复了在流式传输时断开蓝牙耳机会导致崩溃的间歇性问题。</li> <li><code>BroadcastSession.onBroadcastQuality</code> 方法现在报告较低的初始广播质量值。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>增加了对包含多个 <code>AudioBufferLists</code> 的 PCM 缓冲区的支持。这对于 USB 麦克风很常见。</li> </ul> <p>包含 Android 1.2.1 版本的更改：提供了新方法和错误修复，可以正确支持表面尺寸和转动更改：</p> <ul style="list-style-type: none"> <li>修复了 <code>SurfaceSource.setSize(...)</code> 未为其设置新尺寸的错误 <code>SurfaceSource</code>。</li> <li>添加了 <code>Device.setRotation(float rotation)</code> 方法，用于以弧度为单位设置设备上的转动。</li> <li>添加了启用/禁用旋转物理手机 <code>ImageDevice</code> 时的自动旋转 <code>ImageDevice.rotateOnConfigurationChanges(boolean enable)</code> 的方法。</li> <li>添加了返回是否配置 <code>ImageDevice</code> 为在物理听筒旋转时自动旋转 <code>ImageDevice.willRotateOnConfigurationChanges()</code> 的方法。</li> </ul>
<p><a href="#">iOS 广播 SDK 1.3.0</a></p>	<p>下载：<a href="https://broadcast.live-video.net/1.3.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.3.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/ios/</a></p> <ul style="list-style-type: none"> <li>修复了使用 <code>createAppBackgroundImageSource</code> 方法时的几个争用条件，这可能会阻止流在应用程序返回前台后恢复。</li> <li>增加了对 arm64 模拟器的支持。</li> </ul>



## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.531 MB	4.411 MB
armeabi-v7a	1.420 MB	3.525 MB
x86_64	1.719 MB	4.877 MB
x86	1.659 MB	4.925 MB

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.30 MB	3.06MB

2022 年 3 月 1 日

## Amazon IVS 播放器 SDK 1.8.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>在播放录制内容时，减少了质量切换期间冻结的发生。</li> </ul>
<a href="#">Web 播放器 1.8.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package</a> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.8.0/amazon-ivs-player.min.js">https://player.live-video.net/1.8.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.8.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.8.0/amazon-ivs-videojs-tech.min.js</a></p>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.8.0/web/</a></p> <ul style="list-style-type: none"> <li>修复了在某些浏览器中播放录制内容可能会停滞的边缘情况。</li> <li>修复了一个问题，即在录制的视频上向前和向后搜索后，没有触发定时元数据事件。</li> <li>删除了有关 <code>remove()</code> 上的 JW 播放器集成的不必要、令人困惑的警告。</li> <li>为提示类型启用了更严格的类型检查，以支持正确的提示类型筛选。</li> </ul>
<a href="#">Android 播放器 1.8.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.8.0/android/</a></p> <ul style="list-style-type: none"> <li>删除了 <code>ViewUtil</code> 类，它是内部类，并且已被弃用。请改用 <code>PlayerView</code>。</li> </ul>
<a href="#">iOS 播放器 1.8.0</a>	<p>下载：<a href="https://player.live-video.net/1.8.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.8.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.8.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.8.0/ios/</a></p>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	0.975 MB	2.761 MB
armeabi-v7a	0.882 MB	2.177 MB
x86_64	1.020 MB	2.603 MB
x86	1.043 MB	2.890 MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.46 MB	1.10 MB

2022 年 2 月 3 日

## Amazon IVS 广播 SDK：Android 1.2.1

平台	下载和更改
<a href="#">Android 广播 SDK 1.2.1</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.2.1/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.2.1/android/</a></p> <p>此版本提供了新的方法和错误修复，可以正确支持表面尺寸和转动更改。这对于涉及自定义视频输入的使用案例来说是必需的。</p> <ul style="list-style-type: none"><li>修复了 <code>SurfaceSource.setSize(...)</code> 未为其设置新尺寸的错误 <code>SurfaceSource</code>。</li><li>添加了 <code>Device.setRotation(float rotation)</code> 方法，用于以弧度为单位设置设备上的转动。</li><li>添加了启用/禁用旋转物理手机 <code>ImageDevice</code> 时的自动旋转 <code>ImageDevice.rotateOnConfigurationChanges(boolean enable)</code> 的方法。</li><li>添加了返回是否配置 <code>ImageDevice</code> 为在物理听筒旋转时自动旋转 <code>ImageDevice.willRotateOnConfigurationChanges()</code> 的方法。</li></ul>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.642MB	4.536MB
armeabi-v7a	1.468MB	3.261MB
x86_64	1.866MB	5.225MB
x86	1.809MB	4.916MB

2022 年 1 月 20 日

## Amazon IVS 播放器开发工具包 1.7.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>修复了从源媒体播放列表播放流时出现卡顿问题。</li> </ul>
<a href="#">Web 播放器 1.7.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产：<a href="https://player.live-video.net/1.7.0/amazon-ivs-player.min.js">https://player.live-video.net/1.7.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产：<a href="https://player.live-video.net/1.7.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.7.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.7.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.7.0/web/</a></p> <ul style="list-style-type: none"> <li>修复了在播放 Amazon IVS 录制的视频后未触发定时元数据事件的问题。</li> <li>修复了在 iOS 移动网络浏览器上不可用直播的播放 URL 时不会发出 ErrorNotAvailable 错误的问题。™</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"> <li>修复了使用 Video.js 包装器调用 <code>dispose()</code> 时的控制台警告问题。</li> <li>修复了由于在播放器实例损坏后尝试访问该实例而导致的多个空引用错误。</li> <li>更新了 <code>setQuality</code> 文档以更明确地说明需要侦听 <code>QUALITY_CHANGED</code> 才能收到成功通知。</li> <li>更新了 <code>setInitialBufferDuration()</code> 文档以说明它不支持 iOS 移动浏览器。</li> </ul> <p>已知问题：在查看器快进录制内容后快退时，iOS 浏览器中的定时元数据在快进完成前不会重新触发。例如，如果查看器在开始观看录制的内容时首先快进到 60 秒，然后快退到 30 秒，则在 30 到 60 秒之间不会触发定时元数据。我们预计会在即将发布的版本中解决此问题。</p>
<a href="#">Android 播放器 1.7.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.7.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.7.0/android/</a></p> <ul style="list-style-type: none"> <li>已弃用 <code>ViewUtil</code> 类，这是一个内部类；应使用 <code>PlayerView</code>。该类将在下一个 Amazon IVS 播放器版本（1.8.0，暂定计划在 2022 年第 1 季度发布）中完全移除。</li> <li>增加了 <code>PlayerView.setResizeMode(mode)</code> 以控制视频在视图中的显示方式，允许视频可以选择放大或完全填充视图，忽略视频长宽比。</li> </ul>

平台	下载和更改
<a href="#">iOS 播放器 1.7.0</a>	<p>下载：<a href="https://player.live-video.net/1.7.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.7.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.7.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.7.0/ios/</a></p> <ul style="list-style-type: none"> <li>• iOS SDK 现在需要 iOS 11.0 或更高版本。</li> <li>• SDK 不再包含 arm64e 切片。如果苹果公司将其作为标准架构，则将重新启用。</li> <li>• 修复了在应用程序终止和媒体服务重置事件期间可能发生的罕见崩溃问题。</li> </ul>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.013MB	2.820MB
armeabi-v7a	0.895MB	2.012MB
x86_64	1.119MB	3.099MB
x86	1.125MB	2.970MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	0.46 MB	1.09MB

# 2022 年 1 月 18 日

## R2S3 缩略图配置

此版本允许您启用/禁用直播会话的缩略图录制，并修改了可为直播会话生成缩略图的时间间隔。这是该新功能的第一个版本。请参阅：

- [Amazon IVS 入门](#) – 更新了“第 3 步：使用可选录制创建通道”。
- [自动录制到 Amazon S3](#) – 我们进行了多项更改：
  - 我们在“录制内容”部分增加了一条有关修改 thumbnails 文件夹的说明。
  - 我们增加了一个新的“缩略图”部分。
  - 我们更改了“JSON 元数据文件”部分中有关 thumbnails 和 path 字段的信息。
- [Amazon IVS API 参考](#) – 我们进行了多项更改：
  - RecordingConfiguration 对象中的新字段 (thumbnailConfiguration)。这反过来又会影响 CreateRecordingConfiguration 请求和响应、GetRecordingConfiguration 响应和 GetStreamSession 响应。
  - 新对象: ThumbnailConfiguration.

# 2021 年 12 月 9 日

## Amazon IVS 广播开发工具包 1.2.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>• 通过减少罕见的崩溃事件提高了稳定性。</li> <li>• 添加了新方法，onNetworkHealthChanged (Android) 和 broadcastSession:networkHealthChanged (iOS)。当网络的瞬时质量发生变化时，该方法将提供更新。它可用于提供有关直播何时可能会暂时中断的反馈。</li> <li>• 添加了获取/设置 BroadcastConfiguration.mixer.canvasAspectMode 的</li> </ul>

平台	下载和更改
	<p>方法。当插槽的宽高比模式未明确设置时，这将用作插槽的默认宽高比模式。</p> <ul style="list-style-type: none"><li>更改了 Mixer ( Android ) 和 IVSBroadcastMixer ( iOS ) API :</li><li>添加了 <code>getSlots()</code> ，可返回所有已添加插槽的。</li><li>添加了 <code>unbind</code> ，可将设备从混音器插槽中解除绑定。</li><li>更新了 <code>bind</code>、<code>unbind</code> 和 <code>transition</code> ，可返回一个布尔值，指示成功或失败。</li></ul>
<a href="#">Android 广播开发工具包 1.2.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/android/</a></p> <ul style="list-style-type: none"><li>修复了一个错误：即使启用了透明度，插槽的视频或图像也没有与其下面的其他插槽混合（使用 <code>zIndex</code> 值）。</li></ul>



平台	下载和更改
<a href="#">iOS 广播开发工具包 1.2.0</a>	<p>下载：<a href="https://broadcast.live-video.net/1.2.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.2.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/ios/</a></p> <ul style="list-style-type: none"> <li>• 提高了蓝牙和有线耳机的连接稳定性。</li> <li>• 为 <code>IVSCustomImageSource</code> 增加了对以下像素格式的支持： <ul style="list-style-type: none"> <li>• <code>kCVPixelFormatType_Lossless_420_8_YpCbCr_BiPlanarFullRange</code></li> <li>• <code>kCVPixelFormatType_Lossy_420_8_YpCbCr_BiPlanarFullRange</code></li> <li>• <code>kCVPixelFormatType_Lossless_420_8_YpCbCr_BiPlanarVideoRange</code></li> <li>• <code>kCVPixelFormatType_Lossy_420_8_YpCbCr_BiPlanarVideoRange</code></li> <li>• <code>kcV_lossless_32BGRA_PixelFormatType</code></li> <li>• <code>kcv_lossy_32bgra_PixelFormatType</code></li> </ul> </li> <li>• 修复了使用 <code>createAppBackgroundImageSource</code> 方法时的两个争用条件，这可能会阻止流在应用程序返回前台后恢复。</li> </ul>

## 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.639MB	4.530MB
armeabi-v7a	1.466MB	3.255MB
x86_64	1.863MB	5.219MB

架构	压缩大小	未压缩大小
x86	1.806MB	4.910MB

## 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	1.42MB	3.30MB

2021 年 11 月 23 日

## Amazon IVS 播放器开发工具包 1.6

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>增加了一个新播放器功能“setInitialBufferDuration()”，该功能允许客户设置初始缓冲区持续时间。此持续时间决定何时可以开始播放。允许的范围为 0.1 到 5 秒。此方法对 iOS 浏览器平台没有影响。</li> <li>修复了加载流可能会在网络重新连接期间没有调用 play 方法时播放的错误。</li> <li>修复了过时的隐藏式字幕数据未清除的问题。</li> <li>通过减少罕见的崩溃事件提高了播放器的稳定性。</li> </ul>
<a href="#">Web 播放器 1.6.1</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产：<a href="https://player.live-video.net/1.6.1/amazon-ivs-player.min.js">https://player.live-video.net/1.6.1/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产：<a href="https://player.live-video.net/1.6.1/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.6.1/amazon-ivs-videojs-tech.min.js</a></p>

平台	下载和更改
	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.6.1/web/">https://aws.github.io/amazon-ivs-player-docs/1.6.1/web/</a></p> <ul style="list-style-type: none"> <li>在 <code>setQuality</code> 文档添加了有关视频元素的控件属性如何影响调用的备注。</li> <li>改进了播放器从视频解码和播放列表网络错误中恢复的方式。</li> <li>更改了播放器的默认日志级别（从 <code>warning</code>（警告）到 <code>error</code>（错误）），以匹配其他平台。</li> </ul>
<a href="#">Android 播放器 1.6.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.6.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.6.0/android/</a></p>
<a href="#">iOS 播放器 1.6.0</a>	<p>下载：<a href="https://player.live-video.net/1.6.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.6.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.6.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.6.0/ios/</a></p> <ul style="list-style-type: none"> <li>从下一个 IVS 播放器版本（1.7.0，初步计划在 2022 年第一季度发布）开始，iOS 10 支持将被弃用。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.01MB	2.82MB
armeabi-v7a	0.84 MB	2.16MB
x86_64	1.13MB	2.97MB
x86	1.12MB	3.09MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
通用	0.94MB	2.02MB
arm64	0.47MB	1.11 MB
armv7	0.46 MB	0.89 MB

2021 年 11 月 18 日

## 流运行状况

借助新的高分辨率 CloudWatch 指标，Amazon IVS Stream Health 可让您近乎实时地监控直播的运行状况。您还可以通过两个新的 API 端点深入了解流的事件和输入配置。这是该新功能的第一个版本。请参阅：

- [监控 Amazon IVS 直播流运行状况](#) – 这是新的《Amazon IVS 用户指南》页面。
- [Amazon IVS 入门](#) – 我们更新了“第 2 步：设置 IAM 权限”中的 IAM policy，增加了另外三个 IVS 权限 (GetStream、GetStreamSession、ListStreamSessions) 和 cloudwatch:GetMetricData。
- 使用亚马逊监控亚马逊 IVS CloudWatch — 我们添加了四个新的高分辨率指标 ( IngestAudioBitrateIngestFramerate、IngestVideoBitrate、和KeyframeInterval )。

2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring [IVS 低延迟](#) 流媒体。

- [将 Amazon EventBridge 与 Amazon IVS 配合使用](#) — 我们添加了两个事件：“会话已创建”和“会话已结束”。
- [Amazon IVS API 参考](#) – 多项更改：
  - 两个新端点：GetStreamSession 和 ListStreamSessions。
  - 七个新对象：AudioConfiguration IngestConfiguration、StreamEvent、StreamFilters、StreamSession、StreamSessionSummary、和 VideoConfiguration。
  - 流中的新字段 (streamID) 和 StreamSummary 对象。这反过来又会影响 GetStream 和的 ListStreams 响应。

- ListStreams请求中的新字段 (filtersBy)。

2021 年 10 月 20 日

## Amazon IVS 广播开发工具包 1.1.0 : Android 和 iOS

平台	下载和更改
全部	<ul style="list-style-type: none"><li>• 修复了一个错误，该错误在向转换方法提供的插槽配置的名称与目标插槽名称参数不匹配时，会使混合器插槽配置处于意外状态。</li><li>• 通过减少罕见的崩溃事件提高了稳定性。</li><li>• 重新平衡了预设比特率，以更好地反映预期的用户体验。这些记录在广播开发工具包参考文档中。<ul style="list-style-type: none"><li>• 标准 (纵向/横向) – 初始 : 2.1 Mbps。最大 : 6 Mbps。</li><li>• 基本 (纵向/横向) – 初始 : 1.2 Mbps。最大 : 1.5 Mbps。</li><li>• 游戏 (纵向/横向) (仅限 Android) – 初始 : 2.1 Mbps。最大 : 6 Mbps。</li></ul></li><li>• 增加了对单声道音频的支持。现在可以使用 1 或 2 个音频通道 (分别为单声道或立体声) 配置广播会话。此外，自定义音频源可以配置 1 或 2 个音频通道。</li><li>• 将混合器画布和插槽原点更改为左上角。这对于开发人员来说应该更自然，并且提供了更一致的可用性。如果您使用的是自定义混合器插槽，则必须更新代码；请参阅下面的<a href="#">广播开发工具包混合器：从 1.0.0 迁移到 1.1.0</a>。</li><li>• 增加了一个新的文档页面“<a href="#">广播到 Amazon IVS：混合器指南</a>”。</li></ul>

平台	下载和更改
<a href="#">Android 广播开发工具包 1.1.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/android/</a></p> <ul style="list-style-type: none"><li>• 修复了设备方向更改可能导致开发工具包崩溃的错误。</li><li>• 修复了 <code>getPreviewView()</code> 仅第一次被调用时才起作用的错误。现在，<code>getPreviewView()</code> 在每次被调用时都返回新的 <code>ImagePreviewView</code>，因此，您可以将同一个设备或会话的多个 <code>ImagePreviewViews</code> 同时添加到您的视图层次结构中。请注意，同时使用许多 <code>ImagePreviewViews</code> 可能会降低性能。</li><li>• 添加了 <code>stopSystemCapture()</code>，以在不释放整个广播会话的情况下停止系统捕获服务。</li><li>• 增加了 <code>attachDevice</code> 覆盖，以便在连接设备时忽略混合器插槽首选设备。</li></ul>

平台	下载和更改
<a href="#">iOS 广播开发工具包 1.1.0</a>	<p>下载：<a href="https://broadcast.live-video.net/1.1.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.1.0/AmazonIVSBroadcast.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/ios/</a></p> <ul style="list-style-type: none"> <li>• 设置 <code>IVSMixerSlotConfiguration</code> 对象上的大小现在会将 <code>matchCanvasSize</code> 设置为 <code>false</code>。同样，设置 <code>IVSMixerSlotConfiguration</code> 对象上的宽高比会将 <code>matchCanvasAspectMode</code> 设置为 <code>false</code>。</li> <li>• 增加了对带预编码视频的背景音频的支持。新方法 <code>createAppBackgroundImageSourceOnComplete</code> 会在应用程序处于后台时更改默认的行为。以前，由于开发工具包无法再访问摄像头或 GPU（这意味着无法完成视频输入合成或视频编码），整个流都会停止。</li> </ul> <p>新方法返回了 <code>IVSCustomVideoSource</code> 的子类。通常，<code>IVSCustomVideoSource</code> 允许您提交图像样本进行广播。该子类允许您提交图像样本进行预编码，以便稍后在应用程序处于后台时进行广播。</p>

## 广播开发工具包混合器：从 1.0.0 迁移到 1.1.0

广播开发工具包的 1.1.0 版本改变了混合器坐标系的工作方式。在 1.0.0 版本中，混合器使用了不一致的原点。在 1.1.0 版本中，原点在左上角。查看新的 [Amazon IVS 广播开发工具包混合器指南](#)。

更改画布：水平（X 轴）位置保持不变。与 1.0.0 版本相比，垂直定位倒转。Y 轴值为 0 时，会将插槽放在画布的顶部（而不是像 1.0.0 版本那样放在底部）。为了使插槽保持与 1.0.0 相同的位置，请从画布的高度中减去其当前 Y 值；例如，`config.video.size.height - y`

插槽更改：在 1.1.0 版本中，插槽也具有左上角的原点。方向与 1.0.0 版本保持一致，但原点已从中心移动到左上角。与左上角对齐的插槽将是 (0, 0)，与右下角对齐的插槽是：

$(\text{canvas\_width} - \text{slot\_width}, \text{canvas\_height} - \text{slot\_height})$

要将插槽保持与 1.0.0 相同的位置，请从 X 位置减去一半的宽度，从 Y 位置减去一半的高度。此外，插槽的大小相对于左上角。因此，要从中心扩展插槽，您必须在更改大小的同时更改位置；否则，插槽看起来会向下和向右发展。

### 广播开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	2.23 MB	5.75 MB
armeabi-v7a	2.07 MB	4.99 MB
x86_64	2.35 MB	5.78 MB
x86	2.55 MB	6.78 MB

### 广播开发工具包大小：iOS

架构	压缩大小	未压缩大小
arm64	3.11 MB	6.74 MB

2021 年 9 月 29 日

### Amazon IVS 播放器开发工具包：Android 1.5.1

平台	下载和更改
<a href="#">Android 播放器 1.5.1</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.5.1/android/">https://aws.github.io/amazon-ivs-player-docs/1.5.1/android/</a>



平台	下载和更改
	<ul style="list-style-type: none"> <li>修复了 <code>getVersion()</code> ，现在它将返回正确的版本号。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.00 MB	2.80 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.07 MB
x86	1.12 MB	2.94 MB

2021 年 9 月 28 日

## Amazon IVS 播放器开发工具包 1.5.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>修复了加载流可能会在网络重新连接期间没有调用 <code>play</code> 方法时 <code>play</code> 问题。</li> <li>修复了播放器在流断开连接后停留在 <code>PLAYING</code> 状态，而不是移动到 <code>ENDED</code> 状态的问题。</li> <li>更新了 CEA-608 字幕解析以支持更多的编码器。</li> <li>改进了播放器播放直通内容的能力；即，来自 <code>BASIC</code> 通道的内容和来自 <code>STANDARD</code> 通道的最高质量。</li> </ul>
<a href="#">Web 播放器 1.5.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a>

平台	下载和更改
	<p>脚本资产：<a href="https://player.live-video.net/1.5.0/amazon-ivs-player.min.js">https://player.live-video.net/1.5.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产：<a href="https://player.live-video.net/1.5.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.5.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.5.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.5.0/web/</a></p> <ul style="list-style-type: none"><li>• 改进了播放器从视频解码和播放列表网络错误中恢复的方式。</li><li>• 修复了启用本机 HTML5 控件时实时流无法恢复（或延迟后恢复）的错误。</li><li>• 修复了 <code>getBuffered()</code> 方法在没有加载内容时返回未定义而非预期 <code>{ start: 0, end: 0 }</code> 的问题。</li><li>• 在 Video.js 中添加了对 picture-in-picture 模式的支持。</li><li>• 将播放器的默认日志级别更改为错误，而不是警告。</li></ul>
<a href="#">Android 播放器 1.5.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.5.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.5.0/android/</a></p> <ul style="list-style-type: none"><li>• 修复了 Android 开发工具包 30 仿真器上发生的放大错误。</li><li>• 改进了 <code>PlayerView</code> 视图布局的性能。</li><li>• <code>getVersion()</code> 返回 <code>1.5.0-ivs.rc.2</code> 而非 <code>1.5.0</code>。</li></ul>

平台	下载和更改
<a href="#">iOS 播放器 1.5.0</a>	<p>下载：<a href="https://player.live-video.net/1.5.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.5.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.5.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.5.0/ios/</a></p> <ul style="list-style-type: none"> <li>增加了对 Apple Silicon Macs 上的 iOS 模拟器的支持。</li> <li>在播放器被取消分配之前，播放器的内存堆大小在播放过程中会继续增加，已解决此问题。</li> <li>在视频中存在不良数据时，通过忽略该数据并继续播放（而不是停止播放）改进了播放行为。</li> </ul>

### 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.00 MB	2.80 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.07 MB
x86	1.12 MB	2.94 MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
通用	0.92 MB	1.99 MB
arm64	0.47 MB	1.09 MB

架构	压缩大小	未压缩大小
armv7	0.46 MB	0.87 MB

2021 年 9 月 8 日

## Amazon IVS 播放器软件开发工具包 1.4.1

平台	下载和更改
全部	修正了隐藏式字幕解码器，以处理插入顺序出错的字幕。
<a href="#">Web 播放器 1.4.1</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a>	<p><b>NPM Package</b> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产</b> : <a href="https://player.live-video.net/1.4.1/amazon-ivs-player.min.js">https://player.live-video.net/1.4.1/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产</b> : <a href="https://player.live-video.net/1.4.1/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.4.1/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.4.1/web/">https://aws.github.io/amazon-ivs-player-docs/1.4.1/web/</a></p>
<a href="#">Android 播放器 1.4.1</a>	<b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.4.1/android/">https://aws.github.io/amazon-ivs-player-docs/1.4.1/android/</a>
<a href="#">iOS 播放器 1.4.1</a>	<p><b>下载</b> : <a href="https://player.live-video.net/1.4.1/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.4.1/AmazonIVSPlayer.xcframework.zip</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.4.1/ios/">https://aws.github.io/amazon-ivs-player-docs/1.4.1/ios/</a></p>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.00 MB	2.79 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.06 MB
x86	1.11 MB	2.94 MB

## 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
通用	0.89 MB	1.91 MB
arm64	0.45 MB	1.05 MB
armv7	0.44 MB	0.84 MB

## 2021 年 8 月 13 日

### ListTagsForResource API 端点

我们删除了此端点中对分页的支持；即，`maxResults` 请求字段和 `nextToken` 请求/响应字段。（分页无法正常工作。）

## 2021 年 8 月 10 日

### Amazon IVS 播放器开发工具包 1.4.0

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>修复了一个罕见的问题，即如果在 <code>DURATION_CHANGED</code> 事件或 <code>READY</code> 状态更</li> </ul>

平台	下载和更改
	<p>新之后立刻开始搜寻，则 VOD 播放可能会停滞。</p> <ul style="list-style-type: none"><li>• 修正了使用 ID3 元数据播放流时的内存泄漏问题。</li><li>• 修复了注入字幕可能不正确呈现的边缘情况。</li><li>• 提高了播放器自适应比特率流算法的性能。</li><li>• 通过减少罕见的崩溃事件提高了播放器的稳定性。</li><li>• 当从其他线程而非创建它的线程中访问播放器时，添加了一条日志警告消息。</li><li>• 更新 <code>getLiveLatency()</code> 文档，以更具体地说明从服务器到播放器的延迟计算方式。</li></ul>

平台	下载和更改
<p><a href="#">Web 播放器 1.4.0</a> 和 <a href="#">Video.js 集成</a> 以及 <a href="#">JW 播放器集成</a></p>	<p><b>NPM Package</b> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p><b>脚本资产</b> : <a href="https://player.live-video.net/1.4.0/amazon-ivs-player.min.js">https://player.live-video.net/1.4.0/amazon-ivs-player.min.js</a></p> <p><b>Video.js 科技资产</b> : <a href="https://player.live-video.net/1.4.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.4.0/amazon-ivs-videojs-tech.min.js</a></p> <p><b>参考文档</b> : <a href="https://aws.github.io/amazon-ivs-player-docs/1.4.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.4.0/web/</a></p> <ul style="list-style-type: none"> <li>• 修复了一个边缘情况，其中 TIMED_METADATA 事件在 iOS Safari 浏览器上没有触发。</li> <li>• 提高了播放器在 Firefox 上播放低延迟流式传输时的自适应比特率流算法的性能。</li> <li>• <code>getDuration()</code> 的固定文档，它总是为实时流返回 Infinity。</li> <li>• 修复了在桌面 Safari 中自动播放有时失败的错误。</li> <li>• 修复了在开发人员控制台中报告“无法读取未定义项的 'CollectLogs' 属性”的错误。</li> <li>• Video.js : 增加了对 picture-in-picture 模式的支持。</li> <li>• Web : 添加了一个新方法，<code>setRequestCredentials</code>。这将控制播放器是否向跨域端点发出凭证请求。远程端点需要使用适当的 CORS 响应标头 ( 例如 <code>Access-Control-Allow-Origin</code> , 匹配请求的来源 ) , 且 <code>Access-Control-Allow-Credentials</code> 必须为 <code>true</code>。</li> </ul> <p>此设置将在播放器实例的整个生命周期中保持不变。因此，具有 URL 端点的所有后</p>

平台	下载和更改
<p data-bbox="110 562 409 596"><a href="#">Android 播放器 1.4.0</a></p>	<p data-bbox="857 210 1468 289">续 <code>player.load()</code> 调用应该使用适当的 CORS 标头进行响应。</p> <p data-bbox="857 336 1481 512">此方法对 iOS 浏览器平台没有影响。若要允许 iOS 平台上的凭证跨域请求，用户必须明确允许跨站点跟踪并允许 Cookie；这些都位于设备和相应的浏览器应用程序的设置中。</p> <p data-bbox="828 562 1464 642">参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.4.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.4.0/android/</a></p> <ul data-bbox="828 688 1500 1188" style="list-style-type: none"><li>• 修复了高分辨率纵向视频被视为不受支持的视频的问题，即使设备可以支持该视频。</li><li>• 修复了在某些 Android 设备上无法更改播放速率的问题。</li><li>• 更新了背景视频处理，以便在未设置输出表面时不对内容进行解码。</li><li>• 实施了额外的检查，以便在调用 <code>player.release()</code> 方法之后忽略 SDK 调用。这提高了播放器稳定性。</li><li>• 通过优化减少 Android 库文件大小。</li></ul>



平台	下载和更改
<a href="#">iOS 播放器 1.4.0</a>	<p>下载：<a href="https://player.live-video.net/1.4.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.4.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.4.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.4.0/ios/</a></p> <ul style="list-style-type: none"> <li>在播放器被取消分配之前，播放器的内存堆大小在播放过程中会继续增加，已解决此问题。</li> <li>修复了视频播放（例如 UI 动画或 GIF 图像）上面有动画时的潜在停顿问题。</li> <li>修复了媒体服务重置事件期间可能发生的崩溃。</li> <li>解决了质量切换过程中可能出现的 <code>CMFormatDescriptionRef</code> 的内存泄漏。</li> <li>添加了一条错误消息，如果在主线程之外的其它线程上访问 <code>IVSPlayerView</code> 和 <code>IVSPlayerLayer</code> 类的 IVS 特定属性，则会记录该错误消息。</li> <li>更新了背景视频处理，以便在未设置输出表面时不对内容进行解码。</li> <li>改进了 IOS 开发工具包参考中的文档覆盖范围。</li> <li>通过优化减少 iOS 库文件大小。</li> </ul>

## 移动开发工具包大小：Android

架构	压缩大小	未压缩大小
arm64-v8a	1.00 MB	2.79 MB
armeabi-v7a	0.83 MB	2.15 MB

架构	压缩大小	未压缩大小
x86_64	1.11 MB	3.06 MB
x86	1.11 MB	2.93 MB

### 移动开发工具包大小：iOS

架构	压缩大小	未压缩大小
通用	0.89 MB	1.91 MB
arm64	0.45 MB	1.05 MB
armv7	0.44 MB	0.84 MB

2021 年 7 月 27 日

### Amazon IVS 广播开发工具包：Android 1.0.0 和 iOS 1.0.0

平台	下载和更改
<a href="#">Android 广播 SDK 1.0.0</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/android/">https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/android/</a>
<a href="#">iOS 广播 SDK 1.0.0</a>	下载： <a href="https://broadcast.live-video.net/1.0.0/AmazonIVSBroadcast.xcframework.zip">https://broadcast.live-video.net/1.0.0/AmazonIVSBroadcast.xcframework.zip</a> 参考文档： <a href="https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/ios/">https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/ios/</a>

## 2021 年 6 月 1 日

### Amazon IVS 播放器开发工具包 : Android 1.3.3 和 iOS 1.3.3

平台	下载和更改
Android 和 iOS	修复了高分辨率纵向视频被视为不受支持的问题，尽管设备可能支持该视频。
<a href="#">Android 播放器 1.3.3</a>	参考文档： <a href="https://aws.github.io/amazon-ivs-player-docs/1.3.3/android/">https://aws.github.io/amazon-ivs-player-docs/1.3.3/android/</a>
<a href="#">iOS 播放器 1.3.3</a>	<p>下载：<a href="https://player.live-video.net/1.3.3/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.3.3/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.3.3/ios/">https://aws.github.io/amazon-ivs-player-docs/1.3.3/ios/</a></p> <ul style="list-style-type: none"><li>修复了视频播放（例如 UI 动画或 GIF 图像）上面有动画时的潜在停顿问题。</li><li>修复了媒体服务重置事件期间可能发生的崩溃。</li></ul>

## 2021 年 5 月 19 日

### Amazon IVS 播放器开发工具包 : Android 1.3.2

参考文档：<https://aws.github.io/amazon-ivs-player-docs/1.3.2/android/>

为了提高播放器稳定性，实施了额外的检查，以便在调用 `player.release()` 方法之后忽略 API 调用。

2021 年 5 月 5 日

## Amazon IVS 播放器开发工具包 1.3

平台	下载和更改
全部	<ul style="list-style-type: none"> <li>更新了 TextCue 使用文档的 SDK 文档。请参阅 <a href="#">Amazon IVS 文档登录页面</a> 上最新的播放器开发工具包参考。</li> <li>修复了格式错误的单声道输入流的音频播放问题。</li> <li>修复了在实时 HLS 窗口外播放内容时可能出现的罕见播放错误。</li> <li>改进了播放器的标准 HLS 实时流和录制流播放功能。</li> <li>提高了 getLiveLatency 的精度，特别是确保在加载新流时将其重置为零。</li> <li>改进了 ABR ( 自适应比特率流 ) 算法，以便在网络连接改善时更快地提高视频质量。</li> <li>通过减少罕见的崩溃事件提高了播放器的稳定性。</li> </ul>
<a href="#">Web 播放器 1.3.1</a> 和 <a href="#">Video.js 集成</a> 和 <a href="#">JW Player 集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.3.1/amazon-ivs-player.min.js">https://player.live-video.net/1.3.1/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.3.1/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.3.1/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.3.1/web/">https://aws.github.io/amazon-ivs-player-docs/1.3.1/web/</a></p> <ul style="list-style-type: none"> <li>修复了加载后立即执行的定位调用有时被忽略导致播放器从错误位置开始的问题。</li> </ul>

平台	下载和更改
	<ul style="list-style-type: none"><li>• 修复了在录制内容 ( 也称为 VOD ) 中定位的多个问题。</li><li>• 修复了在不理想网络条件下播放可能失败的问题。</li><li>• 在 iOS 移动 Web 浏览器上添加了对 IVS 定时元数据的支持。</li><li>• 修复了在台式机 Safari 中自动播放有时失败的错误。</li><li>• Web 开发工具包 <code>getVersion</code> 函数不再向播放器版本附加哈希。</li><li>• 修复了定位到缓冲范围的准确起点可能会导致另一个定位前移的问题。</li><li>• 在 macOS Safari 14 及更高版本中启用了低延迟 ABR ( 自适应比特率流 ) 。</li><li>• 通过消除不安全的导入副作用，修复了在服务器上下文中加载播放器的问题。</li><li>• 更改了 <code>amazon-ivs-player</code> NPM 包，使其导出由使用的 <code>LogLevel</code> 枚举。<code>setLogLevel</code></li></ul> <p>注意：Web 播放器 1.3.0 NPM 软件包存在，但不起作用。它在 NPM 上被标记为已弃用。如文档中所述，使用 Web 播放器 1.3.1 或更高版本。</p>

平台	下载和更改
<a href="#">Android 播放器 1.3.0</a>	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.3.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.3.0/android/</a></p> <ul style="list-style-type: none"><li>• 修复了当应用程序针对 Android 11 ( API 级别 30 ) 且用户在蜂窝网络上运行 Android 11 时播放器开发工具包崩溃的问题。</li><li>• 修复了网络恢复问题。现在，网络连接断开时播放会自动暂停，并在连接恢复时恢复播放。使用 <code>onNetworkUnavailable</code> 在中回调 <code>Player.Listener</code> 观察网络状态变化。</li><li>• 修复了在播放 VOD 时无法使用 <code>setControlsEnabled(false)</code> 隐藏播放器控件的问题。</li><li>• 修复了如果客户端应用程序使用旧版本 ( 4.0 之前 )，SDK 可能会崩溃的问题。OkHttp</li><li>• Amazon IVS Android 播放器库从 JCenter 存储库移动到 Maven 中央存储库。</li><li>• 从库中删除了 <code>BuildConfig</code> 版本属性。</li></ul>

平台	下载和更改
<a href="#">iOS 播放器 1.3.0</a>	<p>下载：<a href="https://player.live-video.net/1.3.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.3.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.3.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.3.0/ios/</a></p> <ul style="list-style-type: none"><li>• 修复了单个 HLS 媒体段中音频采样率发生变化时 iOS 开发工具包无法正确处理的问题。这可能会导致意外的内存增加和播放失败，或由于媒体错误而导致崩溃。</li><li>• 修复了网络恢复问题。现在，网络连接断开时播放会自动暂停，并在连接恢复时恢复播放。使用 <code>playerNetworkDidBecomeUnavailable</code> 委托方法观察网络状态变化。</li><li>• 修复了导致 iOS 内存可能随时间推移而增加的问题。</li><li>• 增加了音频硬件问题的正常处理。现在，如果收到媒体服务重置通知 (<code>AVAudioSessionMediaServicesWereReset Notification</code>)，播放会自动暂停。请注意，如果重置时正在播放媒体，则仍可能会出现播放错误。</li><li>• 增加了音频会话中断处理。现在，当音频会话中断开始时，播放会自动暂停。当中断结束时，如果播放器之前正在播放，并且中断选项指示应用程序应恢复播放，则会自动恢复播放。</li></ul>

2021 年 4 月 26 日

## Service Quotas 与 CloudWatch 使用量指标集成

您可以使用通过 CloudWatch CloudWatch 使用量指标主动管理您的服务配额。请参阅 [Amazon IVS 服务限额](#)。

## 2021 年 4 月 13 日

### 新 CloudWatch 指标

CloudWatch 为并发观看次数和并发直播添加了指标。参见使用亚马逊监控亚马逊 IVS。CloudWatch

2023 年 10 月 12 日更新：此 CloudWatch 文档已删除，内容已移至 Monitoring [IVS 低延迟流媒体](#)。

更新了相关服务限额的名称以匹配新指标。请参阅 [服务限额（低延迟流式传输）](#)。

有关“视图”的完整定义，请参阅 [Amazon IVS 词汇表](#)。

## 2021 年 4 月 7 日

### 自动记录到 S3 (R2S3)

借助 Amazon IVS，您现在能够将实时视频内容保存到 Amazon S3。保存的视频稍后可用于编辑或作为 VOD 回放之类的操作。

当您为某个通道启用录制时，该通道的所有实时广播都会存储到您选择的 S3 存储桶中。所有可用的高品质呈现和缩略图图像都会保存下来。您的录制配置也会保存，因此可以轻松地将其重复用于其他通道。

您可以通过 Amazon IVS 控制台或 API 设置录制配置和启用/禁用录制。有关详细信息，请参阅 [IVS 入门](#) 和 [Amazon IVS API 参考](#)。

## 2021 年 1 月 28 日

### Amazon IVS 播放器开发工具包：JW 播放器集成 1.2.0

Amazon IVS 播放器现在与 JW Player 集成。请参阅 [JW 播放器集成](#)。

已知问题：在某些情况下，视频的时长似乎为 00:00，并且在进度条上拖动时播放头不定位。只有在 iPhone 上使用 Safari 浏览器观看混合了 Amazon IVS 实时流和 VOD 的无广告播放列表时，才会发生这种情况。



## 2020 年 12 月 16 日

### Amazon IVS 回放器 : SDK for Android 1.2.1

参考文档 : <https://aws.github.io/amazon-ivs-player-docs/1.2.1/android/>

此版本包含一个修复了以下问题的 Android 播放器修补程序 : 在以前的 Android 播放器开发工具包版本中 , 如果应用程序针对 Android 11 ( API 级别 30 ) , 并且用户在蜂窝网络上运行 Android 11 , 开发工具包会崩溃。

## 2020 年 11 月 23 日

### Amazon IVS 播放器开发工具包 1.2.0

平台	下载和更改
全部	<p>改进了 Amazon IVS 流的检测功能 , 因此指标更准确。</p>
<a href="#">Web 播放器 1.2.0</a> 和 <a href="#">Video.js 集成</a>	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package : https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.2.0/amazon-ivs-player.min.js">https://player.live-video.net/1.2.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.2.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.2.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.2.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.2.0/web/</a></p> <ul style="list-style-type: none"> <li>• 如果某个直播的主播放列表不可用 , 我们现在会 ErrorNotAvailable 为所有网络播放源发出。</li> <li>• 更新了与达到并发查看器 (CCV) 限制相关的错误对应的参考文档。</li> </ul>
<a href="#">Android 播放器 1.2.0</a>	<p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.2.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.2.0/android/</a></p>

平台	下载和更改
	<ul style="list-style-type: none"><li>修复了该 <code>getSessionId</code> 功能在安卓系统上崩溃的问题。</li><li>更新了与达到并发查看器 (CCV) 限制相关的错误对应的参考文档。</li></ul> <p>已知问题：如果应用程序针对 Android 11 ( API 级别 30 )，并且用户在蜂窝网络上运行 Android 11，播放器开发工具包会崩溃。此问题将在下一版本中得到修复。与此同时，我们建议定位之前的 Android API 级别 ( 29 或更低 )。</p>
<a href="#">iOS 播放器 1.2.0</a>	<p>下载：<a href="https://player.live-video.net/1.2.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.2.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.2.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.2.0/ios/</a></p> <ul style="list-style-type: none"><li>修复了切换流 URL 或关闭播放器时内存损坏的潜在来源。</li><li>解决了在开始播放前无法启用最佳音频音高校正时可能导致播放失败的问题。音高校正可在播放速度快于或慢于正常速度的情况下改善音频质量。如果音高校正无法启用或最高质量校正算法不可用，则会记录一条消息，但播放将继续。</li></ul> <p>已知问题：如果单个 HLS 媒体段中音频采样率发生变化，iOS 开发工具包将无法正确处理。这可能会导致意外的内存增加和播放失败，或由于媒体错误而导致崩溃。此问题将在下一个主要 iOS 播放器版本中得到修复。</p>

## 2020 年 11 月 12 日

### 新增事件字段 stream\_id

在多个事件中添加了 stream\_id 字段。请参阅 [将 Amazon EventBridge 与 IVS 配合使用](#)。

## 2020 年 11 月 9 日

### 在控制台添加元数据查看功能

现在可以从 Amazon IVS 控制台查看定时元数据。在 Amazon IVS 用户指南中，参阅将元数据嵌入视频流中新增的[“查看定时元数据”](#)章节。

## 2020 年 10 月 30 日

### CloudFormation Support

亚马逊 IVS 现在支持 AWS CloudFormation。这使得 Amazon IVS 客户能够使用 AWS CloudFormation 创建和管理频道、直播密钥和播放密钥对。

Amazon IVS 支持 CloudFormation 适用于所有提供 Amazon IV S 的 [AWS 区域](#)。要开始使用，请参阅 [Amazon IVS 产品页面](#)或 [AWS CloudFormation 用户指南中的亚马逊 IVS 信息](#)。

## 2020 年 10 月 27 日

### 提高通道、CCV 和 CCB 的数量限制

我们提高了三个服务限额限制：

- 每个亚马逊云科技区域中用户可以创建的最大通道数量从 500 增加到 5000。
- 每个亚马逊云科技区域中所有通道中允许播放实时通道的最大并发查看器数量从 3,000 增加到 15,000。
- 每个亚马逊云科技区域中的最大并发广播（可同时进行流式处理的通道）数量从 30 增加到 100。

提供 Amazon IVS 的[所有区域](#)中均适用此增加数量。要了解更多信息，请参阅 Amazon IVS 用户指南中的 [服务限额（低延迟流式传输）](#)。

## 2020 年 10 月 9 日

### 全新 Service Quotas 和 EventBridge 活动

现在有与采集解析相关的服务配额和 EventBridge 事件。请参阅[服务限额 \(低延迟流式传输\)](#)和将[Amazon EventBridge 与 IVS 配合使用](#)。

### Amazon IVS 播放器：适用于 Web 的开发工具包 1.1.2

NPM Package : [https://www.npmjs.com/package/ amazon-ivs-player](https://www.npmjs.com/package/amazon-ivs-player)

脚本资产 : [https://player.live-video.net/1.1.2/ amazon-ivs-player .min.js](https://player.live-video.net/1.1.2/amazon-ivs-player.min.js)

Video.js 科技资产 : [https://player.live-video.net/1.1.2/ amazon-ivs-videojs-tech .min.js](https://player.live-video.net/1.1.2/amazon-ivs-videojs-tech.min.js)

参考文档 : [https://aws.github.io/ amazon-ivs-player-docs /1.1. 2/web/](https://aws.github.io/amazon-ivs-player-docs/1.1.2/web/)

此版本包含一个 Web 播放器修补程序，它修复了影响使用 Microsoft Edge 的查看器的问题。对于这些查看器，如果为流打开了自动质量模式（即 ABR 生效），则低延迟播放将不起作用；在这些情况下，流播放的延迟更高。

## 2020 年 10 月 7 日

### Amazon IVS 播放器开发工具包 1.1.0

Amazon Interactive Video Service (IVS) 播放器开发工具包使用[语义版本控制](#)。

平台	下载和更改
全部	<ul style="list-style-type: none"><li>修复了播放器的自适应比特率算法可能错误地将质量降至 160p 的问题。</li><li>现在，如果没有可播放的视频质量，播放器会引发错误。</li><li>更新了 VOD 定位行为：当试图超越末尾定位时，播放器不会返回错误，而是会定位到末尾。</li><li>现在，在错误恢复期间用尽所有可用质量后，播放器会引发致命错误。</li></ul>

平台	下载和更改
Web 播放器 1.1.0	<p data-bbox="829 226 1507 310"><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package</a> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p data-bbox="829 352 1461 436">脚本资产 : <a href="https://player.live-video.net/1.1.0/amazon-ivs-player.min.js">https://player.live-video.net/1.1.0/amazon-ivs-player.min.js</a></p> <p data-bbox="829 478 1468 562">Video.js 科技资产 : <a href="https://player.live-video.net/1.1.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.1.0/amazon-ivs-videojs-tech.min.js</a></p> <p data-bbox="829 604 1466 688">参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.1.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.1.0/web/</a></p> <p data-bbox="829 730 980 772">已知问题 :</p> <ul data-bbox="829 814 1507 1818" style="list-style-type: none"><li data-bbox="829 814 1507 951">• 现在 , 如果 Video.js 不可用 , <code>registerIVSQualityPlugin</code> 不会写入 <code>console.error</code> , 而是会引发异常。</li><li data-bbox="829 972 1507 1150">• 现在 , 如果 <code>registerIVSTech</code> 或 <code>registerIVSQualityPlugin</code> 被多次调用 , 第一个调用之后的调用不执行任何操作 ( 不会尝试重新注册 ) 。</li><li data-bbox="829 1171 1507 1255">• 将 <code>registerIVSQualityPlugin</code> 的第一个参数的类型从 <code>VideoJS</code> 更改为 <code>any</code> 。</li><li data-bbox="829 1276 1507 1360">• 删除了对浏览器上下文的依赖项 , 以启用服务器端渲染。</li><li data-bbox="829 1381 1507 1507">• 现在 , 如果浏览器自动暂停以响应取消静音 , 播放器会触发 <code>AUDIO_BLOCKED</code> 事件并恢复静音播放。</li><li data-bbox="829 1528 1507 1818">• 添加了网络连接恢复功能。网络超时不会导致错误状态被发送到客户端应用程序。相反 , 当网络连接断开时 :<ul data-bbox="862 1686 1507 1818" style="list-style-type: none"><li data-bbox="862 1686 1507 1818">• 如果应用程序正在播放 , 播放器库会将 <code>NETWORK_UNAVAILABLE</code> 事件添加到应用程序 , 并且播放器进入“闲置”状态。连接</li></ul></li></ul>

平台	下载和更改
<p>Android 播放器 1.1.0</p>	<p>恢复后，播放器库将继续播放，并且应用程序会收到 PLAYING 事件。</p> <ul style="list-style-type: none"> <li>• 如果应用程序已暂停，NETWORK_UNAVAILABLE 事件不会被发送到应用程序，并且播放器库仍处于“闲置”状态。连接恢复后，播放器库将保持“闲置”状态。</li> <li>• 在任何时候，如果应用程序尝试播放，播放器库都会尝试正常播放。NETWORK_UNAVAILABLE 事件被发送到应用程序，并且播放器进入“闲置”状态。</li> </ul> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.1.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.1.0/android/</a></p> <p>已知问题：如果应用程序针对 Android 11 ( API 级别 30 )，并且用户在蜂窝网络上运行 Android 11，播放器开发工具包会崩溃。此问题将在下一版本中得到修复。与此同时，我们建议定位之前的 Android API 级别 ( 29 或更低 )。</p>
<p>iOS 播放器 1.1.0</p>	<p>下载：<a href="https://player.live-video.net/1.1.0/AmazonIVSPlayer.xcframework.zip">https://player.live-video.net/1.1.0/AmazonIVSPlayer.xcframework.zip</a></p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.1.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.1.0/ios/</a></p> <ul style="list-style-type: none"> <li>• 修复了可能导致崩溃且 UIKit 显示以下消息的问题：“Modifications to the layout engine must not be performed from a background thread after it has been accessed from the main thread” ( 在从主线程访问后，不得从后台线程执行对布局引擎的修改 )。在后台和前台处理应用程序时，可能会发生此情况。</li> </ul>

## 2020 年 9 月 14 日

### 新增事件字段 channel\_name

在多个事件中添加了 channel\_name 字段。请参阅 [将 Amazon EventBridge 与 IVS 配合使用](#)。

## 2020 年 8 月 19 日

### 播放授权 (私有通道)

Amazon IVS 现在为客户提供了创建私有通道的功能，允许客户限制哪观众可以观看他们的流。客户通过在通道上启用播放授权并为授权的播放请求生成已签名 JSON Web 令牌 (JWT) 来控制对视频播放的访问。有关详细信息，请参阅 [设置私有通道](#)。

通道对象中新增加的 authorized 字段指示通道是否为私有通道。请参阅 [Amazon IVS API 参考](#)。

## 2020 年 8 月 11 日

### Amazon IVS 播放器 : SDK for iOS 1.0.6

下载 : <已弃用>

参考文档 : <https://aws.github.io/amazon-ivs-player-docs/1.0.6/ios/>

此版本包含一个 iOS 播放器修补程序，它修复了阻止某些 iOS 播放器应用程序添加到 Apple App Store 的问题。具体而言，构建时启用了 bitcode 的应用程序在上传后无法通过 App Store Connect 验证。

## 2020 年 8 月 5 日

### 将亚马逊 EventBridge 与亚马逊 IVS 配合使用

Amazon IVS EventBridge 活动现已通过亚马逊 EventBridge 控制台提供。参见《亚马逊 IVS 用户指南》[为亚马逊 IVS 创建亚马逊 EventBridge 规则](#)中的“将亚马逊 EventBridge 与亚马逊 IVS 搭配使用”部分。

## 2020 年 7 月 15 日

## 播放器版本 1.0

Amazon Interactive Video Service (IVS) 播放器开发工具包使用[语义版本控制](#)。

平台	下载和更改
全部	<p>已知问题：对于 <code>setAutoMaxQuality</code> 和 <code>setQuality</code> 函数，您提供的质量会正确应用到当前流，但如果加载新流，则不会正确应用。要避免此情况，如果加新流，请在 <code>PlayerState.READY</code> 之后使用新流的质量调用此函数。</p>
Web 播放器 1.0.0	<p><a href="https://www.npmjs.com/package/amazon-ivs-player">NPM Package</a> : <a href="https://www.npmjs.com/package/amazon-ivs-player">https://www.npmjs.com/package/amazon-ivs-player</a></p> <p>脚本资产 : <a href="https://player.live-video.net/1.0.0/amazon-ivs-player.min.js">https://player.live-video.net/1.0.0/amazon-ivs-player.min.js</a></p> <p>Video.js 科技资产 : <a href="https://player.live-video.net/1.0.0/amazon-ivs-videojs-tech.min.js">https://player.live-video.net/1.0.0/amazon-ivs-videojs-tech.min.js</a></p> <p>参考文档 : <a href="https://aws.github.io/amazon-ivs-player-docs/1.0.0/web/">https://aws.github.io/amazon-ivs-player-docs/1.0.0/web/</a></p> <p>已知问题：</p> <ul style="list-style-type: none"> <li>在 iOS 移动浏览器（例如 Safari 或 Chrome）上播放 VOD 时，向后定位会使播放器静音。要避免此情况，请在定位后调用 <code>player.setMuted(false)</code>。</li> <li>在 iOS 移动浏览器上播放 VOD 时，直接选择所需位置时向后定位会间歇性地工作。要避免此情况，请将拖动条拖到所需位置。</li> <li>使用 Video.js 集成在 iOS 移动浏览器上播放 VOD 时，回放按钮无法正常工作。要避免此情况，请在初始化 Video.js 时隐藏回放按钮：<a href="https://docs.videojs.com/tutorial-components.html#play-toggle">https://docs.videojs.com/tutorial-components.html#play-toggle</a>。</li> </ul>



平台	下载和更改
Android 播放器 1.0.0	<p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.0.0/android/">https://aws.github.io/amazon-ivs-player-docs/1.0.0/android/</a></p> <p>已知问题：后台处理和前台处理可能会导致 Android 上 VOD 播放的音频/视频失去同步。</p>
iOS 播放器 1.0.0	<p>下载：&lt;已弃用&gt;</p> <p>参考文档：<a href="https://aws.github.io/amazon-ivs-player-docs/1.0.0/ios/">https://aws.github.io/amazon-ivs-player-docs/1.0.0/ios/</a></p> <p>已知问题：</p> <ul style="list-style-type: none"><li>• 后台处理和前台处理会导致实时播放和 VOD 播放失败。要避免此情况，请在收到 <code>UIApplicationDidEnterBackgroundNotification</code> 时暂停流，在收到 <code>UIApplicationDidBecomeActiveNotification</code> 时恢复播放。</li><li>• iOS 10 设备从后台返回时可能会发生崩溃。要避免这种情况，请先将层的 <code>player</code> 属性设置为 <code>nil</code>，然后再进入后台。</li></ul>

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。