

ALIBABA CLOUD

阿里云

DataV数据可视化
可视化应用管理

文档版本：20220609

 阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置>网络>设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击 确定 。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.可视化应用分组管理	05
2.可视化应用排序管理	07
3.可视化应用搜索管理	08
4.使用模板创建PC端可视化应用	09
5.使用识图创建PC端可视化应用	10
6.使用模板创建移动端可视化应用	14
7.编辑可视化应用	15
8.重命名可视化应用	16
9.分享可视化应用	17
10.复制可视化应用	18
11.删除可视化应用	19
12.预览PC端可视化应用	20
13.预览移动端可视化应用	21
14.发布PC端可视化应用	22
15.发布移动端可视化应用	31
16.DataV分享页Token参数签名校验	41

1. 可视化应用分组管理

项目分组管理功能可以将已经创建的可视化应用进行分组归类，使得可视化应用变得有序且功能性一目了然。本文为您介绍DataV项目分组管理功能的使用方法。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面，单击我的分组右侧的+。



3. 在分组名称编辑框内，输入分组名，完成后单击空白处或者按Enter键确定。

您可以对已经生成的分组进行重命名和删除操作。删除分组后，该分组中的可视化应用将全部回归到未分组状态。



4. 移动鼠标到需要进行分组的可视化应用上，单击移动图标，将可视化应用直接拖动到左侧已创建好的分组栏中。



5. 在弹出的对话框中，单击确定。

分组完成后，可单击分组名称，查看分组中包含的可视化应用项目。



2. 可视化应用排序管理

通过DataV新增的可视化应用排序功能，您可以按照名称、创建时间、修改时间这三种方式对已有的可视化应用进行排序，使可视化应用能够有序地排列展示，便于管理。本文档为您介绍可视化应用排序功能的使用方法，帮助您快速对已有可视化应用进行排序展示。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面，移动鼠标至页面最右侧的按修改时间排序（默认）上方。
3. 在列表中单击选择一种可视化应用排序方式。



可视化应用排序成功后，即可看到所有可视化应用按照您选择的方式进行排序。

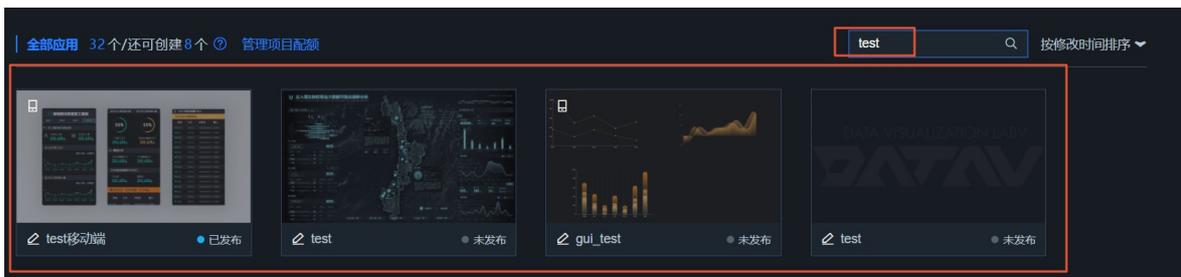
3. 可视化应用搜索管理

通过Dat aV新增的可视化应用搜索功能，您可以根据可视化应用名称快速搜索定位到某个可视化应用。本文为您介绍可视化应用搜索功能的使用方法。

操作步骤

1. 登录Dat aV控制台。
2. 在我的可视化页面右侧的搜索框中，输入您要查找的可视化应用名称。

Dat aV会自动进行匹配搜索，并展示符合搜索条件的可视化应用。



? 说明 搜索功能支持模糊匹配。

4.使用模板创建PC端可视化应用

DataV支持使用模板和识图两种方式创建PC端可视化应用，本文介绍使用模板创建PC端可视化应用的方法。

背景信息

- 基础版的DataV最多可以创建5个可视化应用，企业版最多可以创建20个，专业版最多可以创建40个，尊享版最多可以创建1000个，请根据您的需求合理选择DataV版本。
- 使用识图创建可视化应用的方法请参见[使用识图创建PC端可视化应用](#)。

操作步骤

- 登录DataV控制台。
- 在我的可视化页面内，单击页面上方的PC端创建。



在我的可视化页面的全部应用下方区域，可以查看所创建的所有可视化应用，以及还可以创建的可视化应用的数量。

- 在PC端的模板列表中，选择一个模板，单击创建项目。



- 在创建数据大屏对话框中，输入数据大屏名称，单击创建。
创建成功后，系统会跳转到可视化应用编辑器页面。

说明 您也可以在创建数据大屏对话框中选择大屏分组，前提是已经创建了分组，详情请参见[可视化应用分组管理](#)。

后续步骤

返回我的可视化页面，查看创建成功的可视化应用。

5.使用识图创建PC端可视化应用

DataV支持使用模板和识图两种方式PC端创建可视化应用，本文介绍使用识图创建PC端可视化应用的方法。

背景信息

使用模板创建可视化应用的方法请参见[使用模板创建PC端可视化应用](#)。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面内，单击页面上方的识图创建。



3. 在识图素材选择页面上方，您可以选择如下两种方法选择参考素材，并在完成素材添加后，单击页面右下角的立即生成即可完成素材图片的应用。

选择参考素材样式	说明
使用参考素材	<p>您可以在参考素材模块下方任选其中一个参考图，并单击参考图上的使用参考图，确定选择该参考素材。</p>  <p>The screenshot shows a grid of various dashboard templates under the heading '参考素材'. Each template includes different types of charts, gauges, and data visualizations. A blue button labeled '使用参考图' (Use Reference Image) is visible on one of the templates.</p>

选择参考素材样式	说明
<p>上传图片</p>	<p>在页面上方单击上传图片切换至上传模式。您可以拖拽一张本地图片到上传图片区域虚线范围内，或单击该虚线区域后在本地电脑上传一张自定义图片即可完成添加该参考素材的图片。</p>  <p>您也可以点击图片区域下方单击重新上传按钮，或在单击已上传的图片，对图片进行替换。</p>
<p>使用手绘板</p>	<p>在页面上方单击手绘板切换至手绘模式。您可以在手绘板页面的空白手绘画布区域内，手动绘制所需的组件图像，并一键生成识图可视化应用。</p>  <p>手绘板页面包括如下几个功能：</p> <ul style="list-style-type: none"> ○ 画笔模式：单击空白画布右上角的  图标，在手绘画布中单击鼠标并移动，即可在空白画布中自定义绘制可视化应用组件的图像。 ○ 橡皮模式：单击画布右上角的  图标，在手绘画布中单击鼠标并移动，即可擦除绘制过程中错误的图像。 ○ 清空画布：单击画布右上角的  图标，即可一键清空手绘画布中所有已绘制的图像。 <div style="background-color: #fff9c4; padding: 5px; border: 1px solid #ccc;"> <p> 警告 一键清除后无法恢复内容，请谨慎操作。</p> </div>

4. 进入当前参考图片生成结果页面，您可以在页面选择某个已生成结果样式，一键创建生成识图可视化应用。

当前参考图片生成结果页面包括如下几个功能：

功能	说明
<p>替换当前参考图片</p>	<p>单击当前参考图片区域右下角的替换按钮，可重新上传本地电脑上的任意参考图片。</p>

功能	说明
选择生成结果	<p>每一张被使用的参考图片都会智能生成多个对应的结果样式。</p>  <ul style="list-style-type: none"> 您可以单击预览，即可在页面右侧预览区域查看选中结果样式的效果图。 如果您对当前几个生成结果样式都不满意，可以单击换一批按钮，重新智能生成一批结果样式。
查看上传素材或生成结果	单击页面右上角的  图标，可切换预览区域的内容为上传的素材图片或该素材被选中后智能生成的结果。
打开当前生成结果	单击页面右上角的  图标，可打开并展示当前生成结果的可视化应用内容。
返回上一步	单击页面右下角的返回上一步，可退回素材选择页面，重新选择素材图片或手绘素材图片。

5. 单击当前参考图片生成结果页面右下方的**创建项目**，并输入项目名称和选择项目分组后，Dat aV会一键生成一个识图智能可视化应用。



创建成功后，DataV默认会自动进入创建的可视化应用界面。若没有自动跳转至可视化应用界面，您可以在当前参考图片生成结果页面的**创建成功**对话框中，单击**确定**即可前往当前可视化应用画布编辑页，查看生成的可视化应用内容；单击**取消**即可在此页面继续其他操作。



后续步骤

返回我的可视化页面，查看创建成功的识图可视化应用。

6.使用模板创建移动端可视化应用

DataV支持使用模板方式创建移动端可视化应用，本文介绍使用模板创建移动端可视化应用的方法。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面内，单击页面上方的移动端创建。



在我的可视化页面的全部应用下方区域，可以查看所创建的所有可视化应用，以及还可以创建的可视化应用的数量。

3. 在移动端模板列表中，选择一个模版或空白画板，单击创建项目。



4. 在创建移动屏对话框中，输入数据大屏名称，单击创建。
创建成功后，系统会跳转到移动端可视化应用编辑器页面。

说明 您也可以在创建数据大屏对话框中选择大屏分组，前提是已经创建了分组，详情请参见[可视化应用分组管理](#)。

后续步骤

返回我的可视化页面，查看创建成功的移动端可视化应用。

7. 编辑可视化应用

本文档介绍编辑可视化应用的方法，帮助您快速修改可视化应用的内容和配置。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个可视化应用，单击编辑图标。



3. 在可视化应用编辑页面，修改组件的布局 and 配置，或者添加、删除组件，完成可视化应用的编辑。
编辑完成后，可[预览或发布可视化应用](#)，查看效果。

8.重命名可视化应用

本文档介绍如何对一个可视化应用进行重命名，帮助您快速修改可视化应用的名称，便于搜索与管理。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面中，单击对应可视化应用的名称，进入编辑状态，输入新的名称。



[? 说明](#) 可视化应用的名称可以重复。

3. 单击空白处或按Enter键，即可完成可视化应用名称的修改。
可视化应用名称修改成功后，您就可以完成使用名称搜索可视化应用，按名称对可视化应用进行排序等操作。

9.分享可视化应用

本文档介绍如何将您的可视化应用拷贝给其他用户，帮助您快速地将可视化应用分享给他人，实现与他人合作共同完成可视化应用的开发。

警告

- 拷贝过去的可视化应用包含完整的数据配置，为避免数据泄露，在拷贝前请仔细核对用户识别码。
- 拷屏功能只适用于企业版及以上版本。

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个可视化应用，单击拷屏图标。



3. 在拷贝项目给他人对话框中，输入对方的用户识别码（区分大小写）。



获取用户识别码：在我的可视化页面，将鼠标移至用户名称上，即可查看并单击复制用户识别码。



4. 单击拷贝，拷贝当前项目给其他用户。
拷屏完成后，您可以在拷贝项目给他人对话框中的拷屏历史中，查看对方的用户识别码和拷屏时间，帮助您及时了解拷屏动态。

10.复制可视化应用

本文档介绍如何复制可视化应用。通过可视化应用复制功能，您可以在现有可视化应用的基础上，开发类似的可视化应用，或者作为开发测试环境来进行修改更新，不会影响在线生产应用。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个可视化应用，单击复制图标。



复制成功后，系统会自动生成一个名为`xxx_copy`的可视化应用。您可以修改此可视化应用的名称，并在此应用的基础上，修改可视化应用的布局 and 配置。

11.删除可视化应用

本文介绍如何删除一个可视化应用。及时删除不需要的可视化应用，可以释放可视化应用的空间与配额。

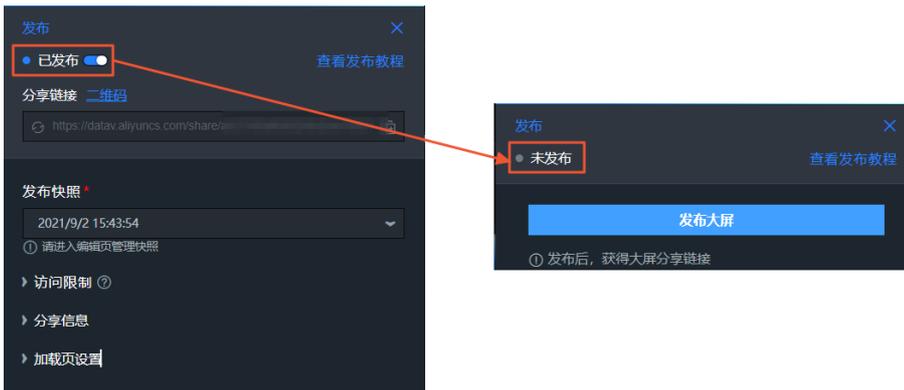
警告 可视化应用删除后不可恢复，请谨慎操作。

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个需要被删除可视化应用。
 - 当被选中删除的可视化应用处于已发布状态，则需要先取消发布才可删除可视化应用，您可以单击可视化应用右上角的发布按钮，在弹窗的发布对话框中，单击关闭已发布开关。关闭发布可视化应用后，单击删除图标，在弹窗出来的二次确认删除对话框中，单击右下角的确定即可删除该可视化应用。

打开已发布可视化应用发布框



关闭已发布开关



- 当被选中删除的可视化应用处于未发布状态，可直接单击删除图标，在弹窗出来的二次确认删除对话框中，单击右下角的确定即可删除该可视化应用。



12. 预览PC端可视化应用

本文介绍如何预览一个PC端可视化应用，帮助您及时查看开发完成的PC端可视化应用效果，便于PC端可视化应用的修改和完善。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个PC端可视化应用，单击右上角的预览图标。



- 对于已发布的可视化应用，如果在发布时设置了访问限制，例如开启了密码访问或Token验证，请联系大屏管理员获取密钥预览；如果在发布时未设置访问限制，可直接预览。
- 对于未发布的可视化应用，可直接预览。

预览成功且PC端可视化应用符合预期后，可将PC端可视化应用发布到线上环境供其他人员在线观看。

13. 预览移动端可视化应用

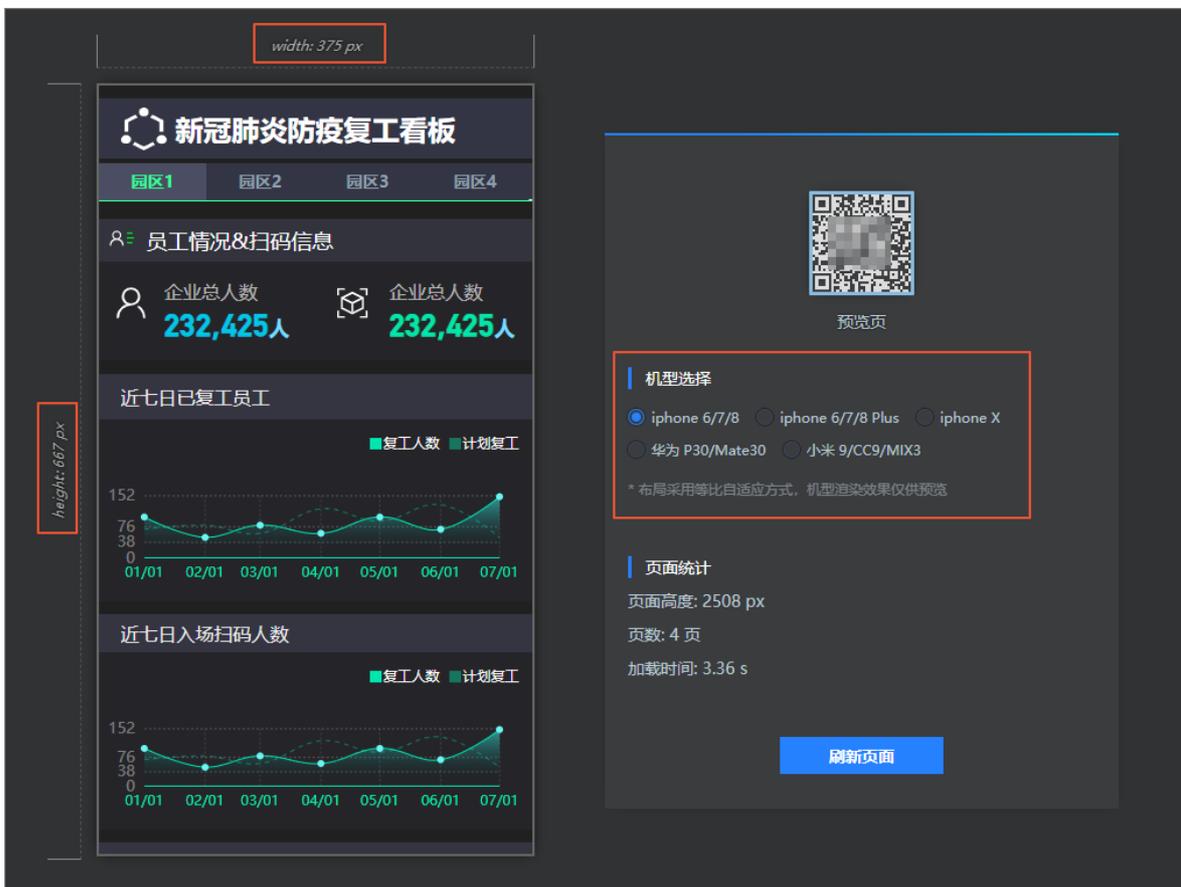
本文介绍如何预览一个移动端可视化应用，帮助您及时查看开发完成的移动端可视化应用效果，便于移动端可视化应用的修改和完善。

操作步骤

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个移动端可视化应用，单击右上角的预览图标。



3. 在移动端的预览界面中，您可以切换不同机型的预览尺寸大小，并在左侧的预览样式页内进行预览。预览页布局采用等比自适应的方式，机型渲染效果仅供预览查看。



4. 您还可以使用手机扫描预览界面右上方的二维码，在您的真机中预览移动端可视化应用的实际效果。
 - 对于已发布的可视化应用，如果在发布时设置了访问限制，例如开启了密码访问或Token验证，请联系大屏管理员获取密钥预览；如果在发布时未设置访问限制，可直接预览。
 - 对于未发布的可视化应用，可直接预览。
5. 在预览界面下方有针对当前预览页面的统计数据，包括可查看到的整个可视化应用的页面高度、在手机页面展示时的页数和加载当前预览页所需的估算时间。
6. 您可以单击刷新页面按钮，刷新当前预览界面。

执行结果

预览成功且移动端可视化应用符合预期后，可将移动端可视化应用发布到线上环境供其他人员在线观看。

14.发布PC端可视化应用

通过PC端可视化应用的发布功能，可以将已经开发完成的PC端可视化应用发布到线上环境，并进行访问权限的设置，供其他人员在线观看。本文介绍如何发布一个PC端可视化应用。

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个可视化应用，单击编辑。
3. 在PC端画布编辑器中，单击页面右上角的发布图标。



4. 在发布对话框中，单击发布大屏。



② 说明 如果是首次发布可视化应用，系统会弹出发布成功的对话框，在对话框中您可以单击前往快照管理查看快照，如果无需查看请单击取消进入发布页面。当首次发布可视化应用时出现发布失败的情况，您可以将当前页面的DataV网址从HTTP协议切换到HTTPS协议网址后再重新选择发布。



发布成功后，系统会开启已发布开关，并生成分享链接和二维码。

5. 扫描二维码访问您的PC端可视化应用。
单击二维码可展示当前可视化应用的二维码，扫描该二维码即可在线访问您的可视化应用。



6. 在线访问您的PC端可视化应用。
单击分享链接右侧的复制图标，打开浏览器，将复制的链接粘贴到地址栏中，即可在线访问您的PC端可视化应用。



说明 可视化应用项目发布后，如果您想更换分享链接，可单击分享链接右侧的 （重新生成链接）图标，重新生成一个分享链接。重新生成后，旧的分享链接将不可用，请使用新的分享链接访问目标可视化应用。

可视化应用分享成功后，您可以配置**发布快照**的参数内容，还可以进行**访问限制**、**分享信息**和**加载页设置**的配置。

注意 为了更好地支持分享，Datav可视化应用分享页域名即将从datav.aliyun.com/share/example切换为datav.aliyuncs.com/share/example，此变更可能对您的Datav可视化应用项目造成的影响，对您造成的不便敬请谅解。请参见**发布分享页域名变更常见问题**及时排查并解决可能存在的问题。

发布快照

设置分享链接后，可以配置**发布快照**，指定访问者看到的可视化应用版本（默认为快照发布版本）。屏幕的内容会锁定在快照创建的那一刻。

注意 存档之后，屏幕内容的编辑和修改不会同步到历史快照中，可作为稳定预览版本的备份。

具体操作方式如下。

- 在**发布**对话框中，在**发布快照**列表中，选择一个已存档的历史快照即可完成该历史快照的发布。
 - 如果当前可视化应用没有历史快照，系统会将当前编辑器的内容作为第一个快照进行发布。
 - 如果当前可视化应用有历史快照，系统会自动发布最新一条快照。
- 单击下方**覆盖已发布快照**，把已发布快照内的可视化应用的内容变成当前编辑页下的内容。
- 单击下方**自动新增快照并发布**，自动新增一个快照并选中新增的快照后立刻发布。
- 单击下方**管理快照**，可在管理快照界面管理多个历史快照，详情请参见**快照管理**。

快照管理

说明 当在管理快照界面内，单击新增快照或修改快照，显示新增或修改快照失败时，您可以将当前页面的Datav网址从HTTP协议切换到HTTPS协议网址后再重新操作。

单击**发布**对话框内的**管理快照**，或者您也可以单击可视化应用编辑器右上角**生成快照**右侧的**管理快照**图标（），然后在**快照管理**对话框内可进行如下快照管理操作。

- 查看额度**：在**快照管理**对话框右上角可以查看管理快照界面内剩余可创建快照的额度。一旦快照数量达到上限后无法新增快照，需要删除不需要的快照。

说明 企业版用户有3个快照额度，专业版和尊享版用户有10个快照额度。

- **新增快照**：在快照管理对话框中单击**新增快照**即可新增一个历史快照。
- **覆盖已发布快照**：单击**覆盖已发布快照**，编辑器的内容覆盖发布页快照原有的内容，即更新发布页内容，并且实时生效。更新内容包括画布编辑器配置、数据源配置、蓝图编辑器配置。



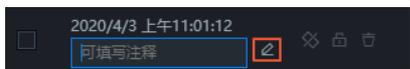
- **发布或关闭快照**：选中管理栏内某个历史快照，单击右侧图标可快捷发布可视化应用或切换发布的快照，切换后，发布页内容将切换成对应快照的内容，再次单击图标即可关闭快照。
- **锁定快照**：单击某个历史快照右侧的图标即可锁定快照，快照锁定后，不可删除和覆盖。
- **批量删除**：在管理界面内多选或全选快照后，单击下方**批量删除**图标可进行快照批量删除。批量删除功能无法删除已发布和被锁定的快照。

说明 已发布的快照无法被删除。



警告 快照目前仅支持切换，不具备回滚的功能，删除后将无法恢复，请谨慎操作。

- **注释快照**：可单击图标，自定义添加快照的注释内容。



访问限制

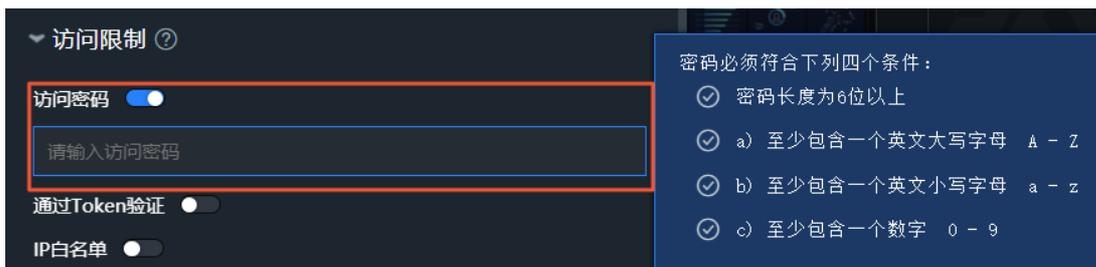
Dat aV的发布功能提供了四种分享可视化应用的方式：

- 公开分享（默认）。
- 密码访问分享。
- 通过Token验证分享。
- IP白名单访问分享。

注意 仅企业版及以上版本支持访问限制功能。

密码访问

1. 在发布对话框中，打开访问密码。



2. 在访问密码输入框中，输入您的验证密码。

密码长度为6位以上，且必须具备以下三个条件：

- 至少包含一个英文大写字母A~Z。
- 至少包含一个英文小写字母a~z。
- 至少包含一个数字0~9。

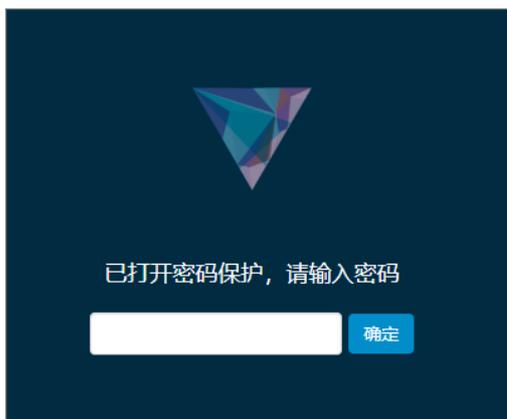
密码设置成功后，系统会提示已设置访问限制。

3. (可选) 配置验证有效期。

注意 只有密码设置成功或开启Token验证后，才可配置验证有效期。

- 选中验证有效期，可以设置密码的有效期，最长为32小时。访问者首次输入密码且成功访问可视化应用后，在设置的有效期内，可任意访问该可视化应用而无需输入密码。
- 去选中验证有效期，每次访问都需要输入密码。

密码设置成功后，当您再次访问可视化应用的分享链接时，系统会提示需要输入密码。



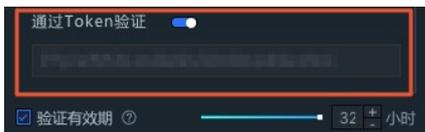
通过Token验证

您可以通过Token验证的方式，将可视化应用的访问权限与您的权限体系进行集成。

在发布对话框中，打开通过Token验证开关，即可开启Token验证。Token验证开启后，您可进行如下操作：

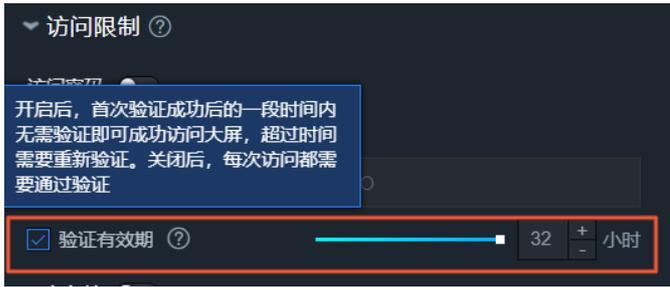
- 获取Token验证码。

打开通过Token验证开关后，DataV会生成一个Token，如下图所示。您需要记录这个Token，以备后用。



- (可选) 配置验证有效期。

- 选中**验证有效期**，可自定义调整Token验证的有效期，最长设置为32小时。访问者首次进行Token验证并成功访问可视化应用后，在设置的有效期时间内，可任意访问该可视化应用而无需再次进行验证。



- 去选中**验证有效期**，每次访问都需要通过验证。

Token验证开启后，再次打开可视化应用的分享页面，会收到一个Access Denied消息，表示您的访问被拒绝了。如果想要打开您的页面，需要完成以下步骤。

注意 为了防止重放攻击，请确保您的服务器时间为东8区标准时间。DataV只会提供1分钟的误差，如果时间误差超过1分钟将会验证失败。

- 发布可视化应用，记录可视化应用编码（URL的最后一段）。
- 将编码与当前时间（毫秒）连起来，并用|（竖线）分隔开。
- 使用Token通过HMAC-SHA256 base64，对上一步得到的Token验证码进行加密。
- 将时间和加密后的签名分别命名为 `_datav_time` 、 `_datav_signature` 。
- 将它们依次放入 `URL` 的querystring中。

注意 如果您的可视化应用URL中需要使用GET的方式传递参数，为了安全性，建议您使用DataV提供的Token参数签名校验，详情请参见[DataV分享页Token参数签名校验](#)。

示例代码如下：

- PHP:

```
<?php
    $token = "kEwoX9rFX9v4zbOT0Gjd_wr65DZ****";
    $screenID = "03d1b68faeb09671046d1ef43f58****";
    $time = time()*1000;
    $stringToSign = $screenID.'|'.$time;
    $signature = urlencode(base64_encode(hash_hmac('sha256', $stringToSign, $token, true)));
    $url = "http://datav.aliyuncs.com/share/".$screenID."?_datav_time=".$time."&_datav_signature=".$signature;
?>
<iframe width=100% height=100% src="<?=$url?>" />
```

- Node.js:

```
const crypto = require('crypto');
var token = "Ev97wOUSAtJusc3Vsd902ngr_vfV****";
var screenID = "14c5448c00ecde02b065c231d165****";
var time = Date.now();
var stringToSign = screenID + '|' + time;
var signature = crypto.createHmac('sha256', token).update(stringToSign).digest().toString('base64');
var url="http://datav.aliyuncs.com/share/"+ screenID +"?_datav_time="+time+"&_datav_signature="+ encodeURIComponent(signature);
```

- Java:

```

package com.company;
import java.security.*;
import java.util.Date;
import javax.crypto.*;
import javax.crypto.spec.SecretKeySpec;
import org.apache.commons.codec.binary.Base64;
import java.net.URLEncoder;

public class TokenTest {
    public static String getSignedUrl(String screenID, String token){
        Date date = new Date();
        Long time = date.getTime();
        String stringToSign = screenID + "|" + time;
        String signature = HMACSHA256(stringToSign.getBytes(), token.getBytes());
        String url = "http://datav.aliyuncs.com/share/"+ screenID +"?_datav_time="+time+"&_datav_signature="+ si
gnature;
        return url;
    }
    /**
     * 使用java原生的摘要实现SHA256加密。
     * @param str 加密后的报文。
     * @return
     */
    public static String HMACSHA256(byte[] data, byte[] key)
    {
        try {
            SecretKeySpec signingKey = new SecretKeySpec(key, "HmacSHA256");
            Mac mac = Mac.getInstance("HmacSHA256");
            mac.init(signingKey);
            return URLEncoder.encode(byte2Base64(mac.doFinal(data)));
        } catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        } catch (InvalidKeyException e) {
            e.printStackTrace();
        }
        return null;
    }
    private static String byte2Base64(byte[] bytes){
        return Base64.encodeBase64String(bytes);
    }
    public static void main(String[] args) throws Exception {
        System.out.println(getSignedUrl("screenId", "token"));
    }
}

```

o C#:

```

using System;
using System.Security.Cryptography;
using System.Text.RegularExpressions;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Text;
namespace datavToken
{
    class Program
    {
        static void Main(string[] args)
        {
            var dic = new Dictionary<string, string>(); // 自定义参数。
            dic.Add("datav_sign_no", "123998"); // datav_sign_开头, 需要签名。
            dic.Add("datav_sign_lo", "kk");
            dic.Add("datav_sign_ao", "xx");
            dic.Add("name", "123"); // 不需要签名。
            // 分享页前缀, 屏幕分享id, token, 自定义参数字典。
            Console.WriteLine(GenerateUrl("https://datav.aliyun.com/share/", "ca74bea5e45503070d607795e0****", "
66DsL2qjrXRHluSJSvcv_f1OUhn****", dic));
        }
        private static string GenerateUrl(string datavBase, string screenId, string token, Dictionary<string, st

```

```
ring> customeParams)
{
    string pattern = @"^datav_sign_.*";
    string timestamp = GetTimeStamp();
    // 参数排序。
    Dictionary<string, string>.KeyCollection keyCol = customeParams.Keys;
    List<string> signKeys = new List<string>();
    foreach (var item in keyCol.ToList())
    {
        if (Regex.IsMatch(item, pattern))
        {
            signKeys.Add(item);
        }
    }
    // 按照key排序。
    signKeys = signKeys.OrderBy(k => k).ToList();
    string paramsSignStr = signKeys.Aggregate("", (total, key) =>
    {
        if (total != "")
        {
            total += "&";
        }
        total += key + "=" + customeParams[key];
        return total;
    });
    //不存在参数认证
    string signStr = screenId + "|" + timestamp;
    //存在参数认证
    //string signStr = screenId + "|" + timestamp + "|" + paramsSignStr;
    var encoding = new System.Text.ASCIIEncoding();
    byte[] keyByte = encoding.GetBytes(token);
    byte[] messageBytes = encoding.GetBytes(signStr);
    string signature;
    using (var hmacsha256 = new HMACSHA256(keyByte))
    {
        byte[] hashmessage = hmacsha256.ComputeHash(messageBytes);
        signature = Convert.ToBase64String(hashmessage);
    }
    var paramDic = new Dictionary<string, string>();
    paramDic.Add("_datav_time", timestamp);
    paramDic.Add("_datav_signature", signature);
    foreach (var item in customeParams)
    {
        paramDic.Add(item.Key, item.Value);
    }
    return datavBase + screenId + "?" + ParseToString(paramDic);
}
public static string GetTimeStamp()
{
    TimeSpan ts = DateTime.UtcNow - new DateTime(1970, 1, 1, 0, 0, 0);
    return Convert.ToInt64(ts.TotalMilliseconds).ToString();
}
static public string ParseToString(IDictionary<string, string> parameters)
{
    IDictionary<string, string> sortedParams = new SortedDictionary<string, string>(parameters);
    IEnumerator<KeyValuePair<string, string>> dem = sortedParams.GetEnumerator();
    StringBuilder query = new StringBuilder("");
    while (dem.MoveNext())
    {
        string key = dem.Current.Key;
        string value = dem.Current.Value;
        if (!string.IsNullOrEmpty(key) && !string.IsNullOrEmpty(value))
        {
            query.Append(key).Append("=").Append(HttpUtility.UrlEncode(value)).Append("&");
        }
    }
    string content = query.ToString().Substring(0, query.Length - 1);
    return content;
}
}
```

IP白名单访问

设置启用IP白名单后，发布后的可视化应用仅允许在设置范围内的白名单IP下访问。

1. 在发布对话框中，打开IP白名单。



2. 在下方IP白名单输入框中，输入您的IP，并用“,”号分隔不同的白名单。

IP白名单设置成功后，当您再次访问可视化应用的分享链接时，如果您使用了非白名单IP访问当前可视化应用，可视化应用页面访问将被拒绝。



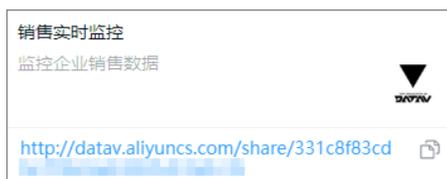
分享信息

DataV可视化应用在钉钉客户端内分享发布链接时提供了可自定义修改分享信息内容的功能，详细参数配置介绍如下。

分享信息配置内容样例



钉钉客户端分享链接样例展示



配置参数	说明
标题	您可以自定义修改即将发布的可视化应用分享链接的标题名称，默认显示为创建该可视化应用时的标题名。

配置参数	说明
描述	您可以对即将发布的可视化应用分享链接进行简单的内容描述。
图片	<p>您可以对即将发布的可视化应用，通过自定义输入图片URL链接或拖动本地图片到右侧图片栏内，为分享链接设置一张图片。</p> <p> 注意 此处支持添加静态或动态图片，但在分享链接展示中均为静态展示。</p>

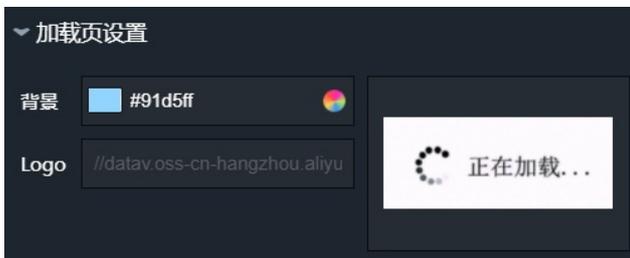
 **注意** 如果您分享的卡片未生效，可能是由于钉钉客户端的缓存机制，导致多次修改分享信息。建议您重启钉钉客户端，再进行链接分享。

加载页设置

Dat aV可视化应用在分享发布链接时提供了可自定义设置加载页内容的功能，详细参数配置介绍如下。

 **注意** 仅专业版及以上版本支持加载页配置功能。

加载页配置内容样例



加载页样例展示



配置参数	说明
背景	您可以自定义修改即将发布的可视化应用加载页的背景颜色，支持纯色及渐变色。
Logo	<p>您可以对即将发布的可视化应用加载页的Logo图片，通过自定义输入图片URL链接或拖动本地图片到右侧图片栏内，为Logo设置一张图片。</p> <p> 注意 此处支持添加静态或动态图片，如果设置的是动态图片，在加载页面会动态展示。</p>

15. 发布移动端可视化应用

通过移动端可视化应用的发布功能，可以将已经开发完成的移动端可视化应用发布到线上环境，并进行访问权限的设置，供其他人员在线观看。本文介绍如何发布一个移动端可视化应用。

1. 登录DataV控制台。
2. 在我的可视化页面中，选择一个移动端可视化应用，单击编辑。
3. 在移动端画布编辑器中，单击页面右上角的发布图标。



4. 在发布对话框中，单击发布大屏。



① 说明 如果是首次发布移动端可视化应用，系统会弹出发布成功的对话框，在对话框中您可以单击前往快照管理查看快照，如果无需查看请单击取消进入发布页面。当首次发布移动端可视化应用时出现发布失败的情况，您可以将当前页面的DataV网址从HTTP协议切换到HTTPS协议网址后再重新选择发布。



发布成功后，系统会开启已发布开关，并生成分享链接和二维码。

5. 扫描二维码访问您的移动端可视化应用。

单击二维码可展示当前移动端可视化应用的二维码，扫描该二维码即可在线访问您的移动端可视化应用。



6. 在线访问您的移动端可视化应用。

单击分享链接右侧的复制图标，打开浏览器，将复制的链接粘贴到地址栏中，即可在线访问您的移动端可视化应用。



说明 移动端可视化应用项目发布后，如果您想更换分享链接，可单击分享链接右侧的（重新生成链接）图标，重新生成一个分享链接。重新生成后，旧的分享链接将不可用，请使用新的分享链接访问目标移动端可视化应用。

移动端可视化应用分享成功后，您可以配置**发布快照**的参数内容，还可以进行**访问限制**、**分享信息**和**加载页设置**的配置。

发布快照

设置分享链接后，可以配置**发布快照**，指定访问者看到的可视化应用版本（默认为快照发布版本）。屏幕的内容会锁定在快照创建的那一刻。

注意 存档之后，屏幕内容的编辑和修改不会同步到历史快照中，可作为稳定预览版本的备份。

具体操作方式如下。

- 在发布对话框中，在**发布快照**列表中，选择一个已存档的历史快照即可完成该历史快照的发布。
 - 如果当前可视化应用没有历史快照，系统会将当前编辑器的内容作为第一个快照进行发布。
 - 如果当前可视化应用有历史快照，系统会自动发布最新一条快照。
- 单击下方**覆盖已发布快照**，把已发布快照内的可视化应用的内容变成当前编辑页下的内容。
- 单击下方**自动新增快照并发布**，自动新增一个快照并选中新增的快照后立刻发布。
- 单击下方**管理快照**，可在管理快照界面管理多个历史快照，详情请参见**快照管理**。

快照管理

说明 当在管理快照界面内，单击新增快照或修改快照，显示新增或修改快照失败时，您可以将当前页面的Datav网址从HTTP协议切换成HTTPS协议网址后再重新操作。

单击发布对话框内的**管理快照**，或者您也可以单击可视化应用编辑器右上角**生成快照**右侧的**管理快照**图标（），然后在**快照管理**对话框内可进行如下快照管理操作。

- 查看额度**：在**快照管理**对话框右上角可以查看管理快照界面内剩余可创建快照的额度。一旦快照数量达到上限后无法新增快照，需要删除不需要的快照。

说明 企业版用户有3个快照额度，专业版和尊享版用户有10个快照额度。

- 新增快照**：在**快照管理**对话框中单击**新增快照**即可新增一个历史快照。
- 覆盖已发布快照**：单击**覆盖已发布快照**，编辑器的内容覆盖发布页快照原有的内容，即更新发布页内容，并且实时生效。更新内容包括画布编辑器配置、数据源配置、蓝图编辑器配置。



- **发布或关闭快照：**选中管理栏内某个历史快照，单击右侧图标可快捷发布可视化应用或切换发布的快照，切换后，发布页内容将切换成对应快照的内容，再次单击图标即可关闭快照。
- **锁定快照：**单击某个历史快照右侧的图标即可锁定快照，快照锁定后，不可删除和覆盖。
- **批量删除：**在管理界面内多选或全选快照后，单击下方**批量删除**图标可进行快照批量删除。批量删除功能无法删除已发布和被锁定的快照。

 说明 已发布的快照无法被删除。



 警告 快照目前仅支持切换，不具备回滚的功能，删除后将无法恢复，请谨慎操作。

- **注释快照：**可单击图标，自定义添加快照的注释内容。



访问限制

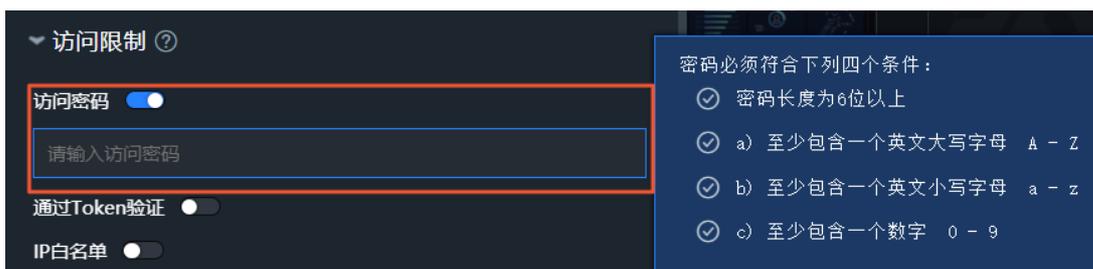
DataV的发布功能提供了四种分享可视化应用的方式：

- 公开分享（默认）。
- 密码访问分享。
- 通过Token验证分享。
- IP白名单访问分享。

 注意 仅企业版及以上版本支持访问限制功能。

密码访问

1. 在发布对话框中，打开访问密码。



2. 在访问密码输入框中，输入您的验证密码。

密码长度为6位以上，且必须具备以下三个条件：

- 至少包含一个英文大写字母A~Z。
- 至少包含一个英文小写字母a~z。
- 至少包含一个数字0~9。

密码设置成功后，系统会提示已设置访问限制。

3. (可选) 配置验证有效期。

注意 只有密码设置成功或开启Token验证后，才可配置验证有效期。

- 选中验证有效期，可以设置密码的有效期，最长为32小时。访问者首次输入密码且成功访问可视化应用后，在设置的有效期时间内，可任意访问该可视化应用而无需输入密码。
- 去选中验证有效期，每次访问都需要输入密码。

密码设置成功后，当您再次访问可视化应用的分享链接时，系统会提示需要输入密码。



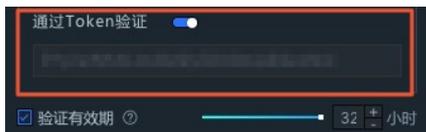
通过Token验证

您可以通过Token验证的方式，将可视化应用的访问权限与您的权限体系进行集成。

在发布对话框中，打开通过Token验证开关，即可开启Token验证。Token验证开启后，您可进行如下操作：

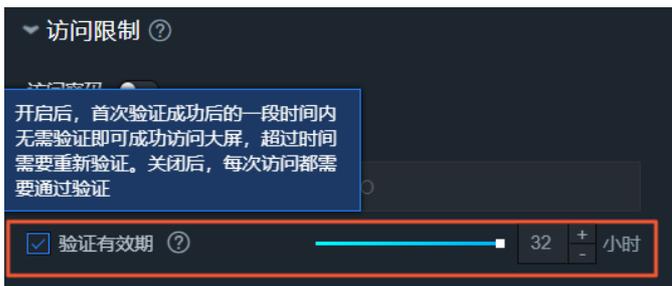
- 获取Token验证码。

打开通过Token验证开关后，DataV会生成一个Token，如下图所示。您需要记录这个Token，以备后用。



- (可选) 配置验证有效期。

- 选中**验证有效期**，可自定义调整Token验证的有效期，最长设置为32小时。访问者首次进行Token验证并成功访问可视化应用后，在设置的有效期时间内，可任意访问该可视化应用而无需再次进行验证。



- 去选中**验证有效期**，每次访问都需要通过验证。

Token验证开启后，再次打开可视化应用的分享页面，会收到一个Access Denied消息，表示您的访问被拒绝了。如果想要打开您的页面，需要完成以下步骤。

注意 为了防止重放攻击，请确保您的服务器时间为东8区标准时间。DataV只会提供1分钟的误差，如果时间误差超过1分钟将会验证失败。

- 发布可视化应用，记录可视化应用编码（URL的最后一段）。
- 将编码与当前时间（毫秒）连起来，并用|（竖线）分隔开。
- 使用Token通过HMAC-SHA256 base64，对上一步得到的Token验证码进行加密。
- 将时间和加密后的签名分别命名为 `_datav_time` 、 `_datav_signature` 。
- 将它们依次放入 `URL` 的querystring中。

注意 如果您的可视化应用URL中需要使用GET的方式传递参数，为了安全性，建议您使用DataV提供的Token参数签名校验，详情请参见[DataV分享页Token参数签名校验](#)。

示例代码如下：

- PHP:

```
<?php
    $token = "kEwoX9rFX9v4zbOT0Gjd_wr65DZ****";
    $screenID = "03d1b68faeb09671046d1ef43f58****";
    $time = time()*1000;
    $stringToSign = $screenID.'|'.$time;
    $signature = urlencode(base64_encode(hash_hmac('sha256', $stringToSign, $token, true)));
    $url = "http://datav.aliyuncs.com/share/".$screenID."?_datav_time=".$time."&_datav_signature=".$signature;
?>
<iframe width=100% height=100% src="<?=$url?>" />
```

- Node.js:

```
const crypto = require('crypto');
var token = "Ev97wOUSAtJusc3Vsd902ngr_vfV****";
var screenID = "14c5448c00ecde02b065c231d165****";
var time = Date.now();
var stringToSign = screenID + '|' + time;
var signature = crypto.createHmac('sha256', token).update(stringToSign).digest().toString('base64');
var url="http://datav.aliyuncs.com/share/"+ screenID +"?_datav_time="+time+"&_datav_signature="+ encodeURIComponent(signature);
```

- Java:

```

package com.company;
import java.security.*;
import java.util.Date;
import javax.crypto.*;
import javax.crypto.spec.SecretKeySpec;
import org.apache.commons.codec.binary.Base64;
import java.net.URLEncoder;

public class TokenTest {
    public static String getSignedUrl(String screenID, String token){
        Date date = new Date();
        Long time = date.getTime();
        String stringToSign = screenID + "|" + time;
        String signature = HMACSHA256(stringToSign.getBytes(), token.getBytes());
        String url = "http://datav.aliyuncs.com/share/"+ screenID +"?_datav_time="+time+"&_datav_signature="+ signature;
        return url;
    }
    /**
     * 使用java原生的摘要实现SHA256加密。
     * @param str 加密后的报文。
     * @return
     */
    public static String HMACSHA256(byte[] data, byte[] key)
    {
        try {
            SecretKeySpec signingKey = new SecretKeySpec(key, "HmacSHA256");
            Mac mac = Mac.getInstance("HmacSHA256");
            mac.init(signingKey);
            return URLEncoder.encode(byte2Base64(mac.doFinal(data)));
        } catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        } catch (InvalidKeyException e) {
            e.printStackTrace();
        }
        return null;
    }
    private static String byte2Base64(byte[] bytes){
        return Base64.encodeBase64String(bytes);
    }
    public static void main(String[] args) throws Exception {
        System.out.println(getSignedUrl("screenId", "token"));
    }
}

```

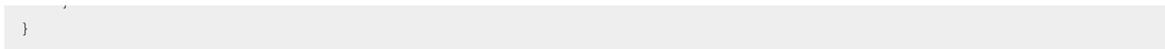
o C#:

```

using System;
using System.Security.Cryptography;
using System.Text.RegularExpressions;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Text;
namespace datavToken
{
    class Program
    {
        static void Main(string[] args)
        {
            var dic = new Dictionary<string, string>(); // 自定义参数。
            dic.Add("datav_sign_no", "123998"); // datav_sign_开头, 需要签名。
            dic.Add("datav_sign_lo", "kk");
            dic.Add("datav_sign_ao", "xx");
            dic.Add("name", "123"); // 不需要签名。
            // 分享页前缀, 屏幕分享id, token, 自定义参数字典。
            Console.WriteLine(GenerateUrl("https://datav.aliyun.com/share/", "ca74bea5e45503070d607795e0****", "66DsL2qjrXRHluSJSvcv_f1OUhn****", dic));
        }
        private static string GenerateUrl(string datavBase, string screenId, string token, Dictionary<string, st

```

```
ring> customeParams)
{
    string pattern = @"^datav_sign_.*";
    string timestamp = GetTimeStamp();
    // 参数排序。
    Dictionary<string, string>.KeyCollection keyCol = customeParams.Keys;
    List<string> signKeys = new List<string>();
    foreach (var item in keyCol.ToList())
    {
        if (Regex.IsMatch(item, pattern))
        {
            signKeys.Add(item);
        }
    }
    // 按照key排序。
    signKeys = signKeys.OrderBy(k => k).ToList();
    string paramsSignStr = signKeys.Aggregate("", (total, key) =>
    {
        if (total != "")
        {
            total += "&";
        }
        total += key + "=" + customeParams[key];
        return total;
    });
    //不存在参数认证
    string signStr = screenId + "|" + timestamp;
    //存在参数认证
    //string signStr = screenId + "|" + timestamp + "|" + paramsSignStr;
    var encoding = new System.Text.ASCIIEncoding();
    byte[] keyByte = encoding.GetBytes(token);
    byte[] messageBytes = encoding.GetBytes(signStr);
    string signature;
    using (var hmacsha256 = new HMACSHA256(keyByte))
    {
        byte[] hashmessage = hmacsha256.ComputeHash(messageBytes);
        signature = Convert.ToBase64String(hashmessage);
    }
    var paramDic = new Dictionary<string, string>();
    paramDic.Add("_datav_time", timestamp);
    paramDic.Add("_datav_signature", signature);
    foreach (var item in customeParams)
    {
        paramDic.Add(item.Key, item.Value);
    }
    return datavBase + screenId + "?" + ParseToString(paramDic);
}
public static string GetTimeStamp()
{
    TimeSpan ts = DateTime.UtcNow - new DateTime(1970, 1, 1, 0, 0, 0);
    return Convert.ToInt64(ts.TotalMilliseconds).ToString();
}
static public string ParseToString(IDictionary<string, string> parameters)
{
    IDictionary<string, string> sortedParams = new SortedDictionary<string, string>(parameters);
    IEnumerator<KeyValuePair<string, string>> dem = sortedParams.GetEnumerator();
    StringBuilder query = new StringBuilder("");
    while (dem.MoveNext())
    {
        string key = dem.Current.Key;
        string value = dem.Current.Value;
        if (!string.IsNullOrEmpty(key) && !string.IsNullOrEmpty(value))
        {
            query.Append(key).Append("=").Append(HttpUtility.UrlEncode(value)).Append("&");
        }
    }
    string content = query.ToString().Substring(0, query.Length - 1);
    return content;
}
}
```



IP白名单访问

设置启用IP白名单后，发布后的可视化应用仅允许在设置范围内的白名单IP下访问。

1. 在发布对话框中，打开IP白名单。



2. 在下方IP白名单输入框中，输入您的IP，并用“,”号分隔不同的白名单。

IP白名单设置成功后，当您再次访问可视化应用的分享链接时，如果您使用了非白名单IP访问当前可视化应用，可视化应用页面访问将被拒绝。



分享信息

Datav移动端可视化应用在钉钉客户端内发布时提供了可自定义修改分享信息内容的功能，详细参数配置介绍如下。

分享信息配置内容样例



钉钉客户端内分享链接样例展示



配置参数	说明
标题	您可以自定义修改即将发布的移动端可视化应用分享链接的标题名称，默认显示为创建该移动端可视化应用时的标题名。

配置参数	说明
描述	您可以对即将发布的移动端可视化应用分享链接进行简单的内容描述。
图片	<p>您可以对即将发布的移动端可视化应用，通过自定义输入图片URL链接或拖动本地图片到右侧图片栏内，为分享链接设置一张图片。</p> <p> 注意 此处支持添加静态或动态图片，但在分享链接展示中均为静态展示。</p>

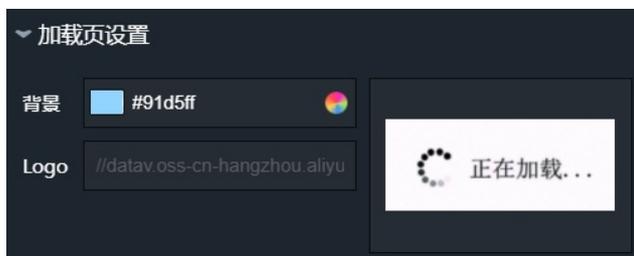
 **注意** 如果您分享的卡片未生效，可能是由于钉钉客户端的缓存机制，导致多次修改分享信息。建议您重启钉钉客户端，再进行链接分享。

加载页设置

DataV可视化应用在分享移动端发布链接时提供了可自定义设置加载页内容的功能，详细参数配置介绍如下。

 **注意** 仅专业版及以上版本支持加载页配置功能。

加载页配置内容样例



移动端加载页样例展示



配置参数	说明
背景	您可以自定义修改即将发布的移动端可视化应用加载页的背景颜色，支持纯色及渐变色。

配置参数	说明
Logo	<p>您可以对即将发布的移动端可视化应用加载页的Logo图片，通过自定义输入图片URL链接或拖动本地图片到右侧图片栏内，为Logo设置一张图片。</p> <p> 注意 此处支持添加静态或动态图片，但在分享链接展示中均为静态展示。</p>

16.DataV分享页Token参数签名校验

本文档介绍在发布DataV大屏时，使用Token参数签名校验的方法。通过Token参数签名校验功能，您可以对大屏交互时传递的参数进行签名鉴权，保证大屏的URL访问链接的参数不能被篡改，从而提高大屏数据以及用户信息的安全性。

前提条件

在使用Token参数签名校验前，请确保：

- 大屏使用Token验证的方式进行发布，具体请参见[通过Token验证](#)。
- 大屏以Get的方式在URL中传递参数（直接在URL后面加参数）。
- 大屏URL中传递的参数要求不能被篡改。

背景信息

下文以一个示例说明Token参数签名校验的使用背景。

某用户的系统嵌入了DataV大屏，URL通过Token计算，通过Get方式传递用户的工号给大屏展示相对应的数据，可以使用 `https://datav.aliyuncs.com/share/xxx?_datav_time=1556022195845&_datav_signature=%2BDZFj3QDIla%2F00fBZLdJMgk2Z1Ocs9MLL1*****%3D&workid=123` 来访问大屏。

其中workid（工号）为大屏传递的参数，存在被篡改的可能。比如工号为123的员工将URL改成 `https://datav.aliyuncs.com/share/xxx?_datav_time=1556022195845&_datav_signature=%2BDZFj3QDIla%2F00fBZLdJMgk2Z1Ocs9MLL1*****%3D&workid=124`，就可以看到工号为124的员工的资料。因此需要对用户传递的参数进行签名鉴权，保证计算得到的URL的参数不能被更改，如果私自更改了传参，页面将无法访问。

签名参数规则

需要加入签名的参数，其参数名需以 `_datav_sign_` 开头，后面可以带任何有效的参数名字符。由此可得此参数名的正则表达式为 `/^_datav_sign_.*`。

 说明 不符合签名参数规则的参数，将不会进行参数签名校验，允许修改参数值。签名参数按升序排序。

带签名参数的URL计算

Node.js代码示例如下。

```
const crypto = require('crypto');
const querystring = require('querystring');
const signedQueryParamReg = /^_datav_sign_.*;/; // 符合此正则表达式的参数是需要签名的。
const token = "93TWmeBtxxxxxxxxxx3thGyAgzennsS";
const screenID = "b92xxxxxxxxxxxxxxxxxx27b4c538cd4";
const time = Date.now();
const customeParams = {
  _datav_sign_no: 123998,
  name: 123
};
let signParamsStr = Object.keys(customeParams)
  .filter(paramName => customeParams[paramName] && signedQueryParamReg.test(paramName))
  .sort()
  .map(param => `${param}=${customeParams[param]}`)
  .join('&');
let stringToSign = [screenID, time];
signParamsStr && stringToSign.push(signParamsStr);
stringToSign = stringToSign.join('|');
let signature = crypto.createHmac('sha256', token).update(stringToSign).digest().toString('base64');
let queryParams = {
  _datav_time: time,
  _datav_signature: signature
};
Object.keys(customeParams).forEach(paramName => {
  queryParams[paramName] = customeParams[paramName];
});
let url = `https://datav.aliyuncs.com/share/${screenID}?${querystring.stringify(queryParams)}`;
console.log(url);
```

使用以上代码示例得到的URL为：`https://datav.aliyuncs.com/share/b92db8e09358c82efca0727b4c538cd4?_datav_time=1556023246894&_datav_signature=GGsbvx1emUeBoRVco8JgrJVWRcmao7NuRYt20*****%3D&datav_sign_no=123998&name=123`。在URL的有效期内，如果修改了datav_sign_no字段的值，链接将无法访问；如果修改了name字段的值，链接仍然可以访问，因为datav_sign_no符合[签名参数规则](#)，参与了签名计算，而name不符合签名参数规则，不会进行签名计算。

使用流程

1. 确定需要签名计算的参数名（即不允许被篡改的参数）。
2. 在大屏开发完成后，使用Token验证的方式发布大屏，具体请参见[通过Token验证](#)。
3. 参见[带签名参数的URL计算](#)，计算大屏的URL。
4. 使用上一步中计算得到的URL访问大屏，在大屏访问过程中，系统会自动进行参数签名校验。

如果参数签名校验功能正常，当您修改了签名参数，再次访问此URL时，访问会被拒绝。