# 阿里云

应用高可用服务 故障演练

文档版本: 20220506

**(一)** 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。	
□ 注意	用于警示信息、补充说明等,是用户必须 了解的内容。	<b>八)注意</b> 权重设置为0,该服务器不会再接受新请求。
⑦ 说明	用于补充说明、最佳实践、窍门等 <i>,</i> 不是用户必须了解的内容。	② 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 <b>结果确认</b> 页面,单击 <b>确定</b> 。
Courier字体	命令或代码。	执行 cd /d C:/window 命令,进入 Windows系统文件夹。
斜体	表示参数、变量。	bae log listinstanceid  Instance_ID
[] 或者 [a b]	表示可选项,至多选择一个。	ipconfig [-all -t]
{} 或者 {a b}	表示必选项,至多选择一个。	switch {active stand}

# 目录

1.什么是故障演练	07
2.功能更新记录	09
3.产品架构	11
4.产品优势	13
4.1. 功能优势	13
4.2. 围绕混沌工程的平台实践	14
4.3. 开源ChaosBlade与商业化AHAS CHAOS故障演练对比	17
5.安装探针	20
5.1. 安装探针概述	20
5.2. 接入ECS	20
5.3. 接入自建Kubernetes集群	22
5.4. 接入容器服务K8s版	24
5.5. 接入主机	25
6.开始演练	27
6.1. 演练概述	27
6.2. 接入演练应用	27
6.3. 安装故障演练探针	28
6.4. 配置RAM权限	29
6.5. 创建演练	31
6.6. 执行演练	34
6.7. 停止演练	36
6.8. 接入ARMS监控	36
7.可视化演练	39
7.1. 什么是可视化演练	39
7.2. 使用可视化演练	39
8.演练场景	43

8.1. 演练场景说明	43
8.2. 基础资源类场景	
8.2.1. CPU类场景	43
8.2.2. 网络类场景	- 44
8.3. Java 类	
8.3.1. 虚拟机场景	
8.3.2. 代码逻辑场景	
8.3.3. JVM注入动态脚本	
8.4. Kubernetes 类场景	
8.4.1. Node演练场景	
8.4.2. Pod演练场景	
8.4.3. Container演练场景	
9.强弱依赖治理	57
9.1. 强弱依赖治理概述	57
9.2. 应用强弱依赖治理	60
9.3. 微服务强弱依赖治理	62
10.消息演练	67
10.1. 什么是消息演练	67
11.容灾演练	- 69
11.1. 同城容灾演练产品概述	69
11.2. 同城容灾架构概述	71
11.3. 容灾演练断网方式说明	72
12.容器演练	- 74
12.1. 什么是容器报警演练	74
12.2. 报警演练	
13.演练空间	
13.1. 演练空间概述	
13.2. 管理演练空间	
13.4. 旨垤澳练工问	//

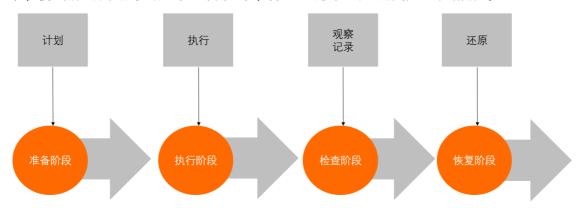
应用高可用服务 故障演练·目录

# 1.什么是故障演练

故障演练是一款遵循混沌工程实验原理并融合了阿里巴巴内部实践的产品,提供丰富故障场景,能够帮助分布式系统提升容错性和可恢复性。

### 流程

故障演练建立了一套标准的演练流程,包含准备阶段、执行阶段、检查阶段和恢复阶段。通过四阶段的流程,覆盖用户从计划到还原的完整演练过程,并通过可视化的方式清晰的呈现给用户。



### 适用场景

故障演练可适用于以下典型场景:

#### ● 衡量微服务的容错能力

通过模拟调用延迟、服务不可用、机器资源满载等,查看发生故障的节点或实例是否被自动隔离、下线,流量调度是否正确,预案是否有效,同时观察系统整体的QPS或RT是否受影响。在此基础上可以缓慢增加故障节点范围,验证上游服务限流降级、熔断等是否有效。最终故障节点增加到请求服务超时,估算系统容错红线,衡量系统容错能力。

### ● 验证容器编排配置是否合理

通过模拟杀服务Pod、杀节点、增大Pod资源负载,观察系统服务可用性,验证副本配置、资源限制配置以及Pod下部署的容器是否合理。

#### ● 测试PaaS层是否健壮

通过模拟上层资源负载,验证调度系统的有效性;模拟依赖的分布式存储不可用,验证系统的容错能力;模拟调度节点不可用,测试调度任务是否自动迁移到可用节点;模拟主备节点故障,测试主备切换是否正常。

### ● 验证监控告警的时效性

通过对系统注入故障,验证监控指标是否准确,监控维度是否完善,告警阈值是否合理,告警是否快速, 告警接收人是否正确,通知渠道是否可用等,提升监控告警的准确和时效性。

### ● 定位与解决问题的应急能力

通过故障突袭,随机对系统注入故障,考察相关人员对问题的应急能力,以及问题上报、处理流程是否合理,达到以战养战,锻炼人定位与解决问题的能力。

### 故障演练与AHAS服务体系

故障演练作为AHAS的一部分,与AHAS其他功能组成了一套完善的高可用保障服务,可以帮助用户实现包括架构、业务、人员的全面高可用提升。故障演练在其中承担着问题发现、问题验证、高可用经验沉淀的作用。



# 2.功能更新记录

本文介绍故障演练Chaos历次发布涉及的功能变更,帮助您了解故障演练的发布动态。

### 2021年10月

功能名称	功能概述	支持资源包类型	支持地域	发布时间
微服务强弱依赖支 持Dubbo	在理务能可赖的Dubbok。 一定, 一定, 一定, 一定, 一定, 一定, 一定, 一定,	专业包	全部地域	2021-10-12

### 2021年9月

功能名称	功能概述	支持资源包类型	支持地域	发布时间
可视化演练	可架外性的面的人类的 医人名	企业包	全部地域	2021-09-15

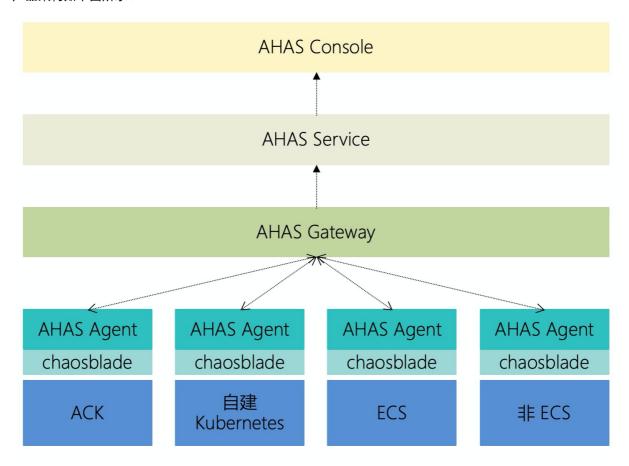
功能名称	功能概述	支持资源包类型	支持地域	发布时间
资源包规格调整	取消入门包,保留 专业包和企业包, 并调整部分包的功 能。已购买的资资。 包权益保持练资。 更多故障演练,请参 见故障演练计费明 细。	-	全部地域	2021-09-15
故障场景参数分层	将数数的量量的量量量数的量量量数的量量的更多的更多的更多的更多的更多的更多的更多的更多的更多的更多的更多的更多的更多的	全部规格	全部地域	2021-09-08
混沌工程动手实验 室	通过免费云资源和 真实云环境,帮助 您了解混沌工程的 原理,体验故障演 练。体验地 址:Chaos带你快速 上手混沌工程。	-	-	2021-09-06

### 2021年8月

功能名称	功能概述	支持资源包规格	支持地域	发布时间
容灾演练	容区交換 网络大型 医克克斯 医克克斯 医克克斯 医克克斯 医克克斯 医克克斯 医克克斯 医克克	企业包	全部地域	2021-08-15

# 3.产品架构

本文向您介绍故障演练的产品架构,以下简称故障演练为 AHAS Chaos。 产品架构如下图所示:



### **AHAS Agent**

AHAS Agent 安装在指定的目标机器上,用来执行服务端下发的故障注入命令以及采集演练相关的必要信息,比如 CPU、内存占用等,主要有以下特点:

- 快捷高效
  - 。 支持在控制台一键安装 AHAS Agent, 如果是公网则需要用户手动安装。
- 安全可靠
  - Agent 本身对应用无入侵。除了演练的故障场景需要作用于应用进程的情况例外,例如 Java 类的故障场景。
  - 网络异常可自动重试。
  - 数据传输通道严格加密,保证数据安全。
- 完善的自我保护
  - 实时监控系统资源,不占用过多带宽。
  - 非用户手动卸载情况下的异常退出可以自我保活,保证 AHAS Agent 在主进程消失的情况下, 可以自动重启。

### ChaosBlade

ChaosBlade 是 AHAS Agent 的核心组件,用来解析、校验和执行服务端下发的故障指令,并且已正式对外开源,具有以下特点:

### ● 简单易用

- 清晰易懂的混沌工程实验模型,易于理解。
- 自带完善的命令行工具,方便本地调试。

#### ● 功能稳定

- 在阿里内部被广泛使用,已通过真实环境的严格考验。
- 活跃的开源社区确保了功能的不断迭代和缺陷的快速修复。

#### ● 场景丰富

- 基础资源: CPU、内存、网络、磁盘、进程等实验场景。
- Java 应用:数据库、缓存、消息、JVM 本身、微服务等,还可以指定任意类方法注入各种复杂的实验场景。
- 云原生场景:
  - 节点上 CPU、内存、网络、磁盘、进程实验场景
  - Pod 网络和 Pod 本身实验场景例如删除 Pod
  - 容器的实验场景例如删除容器、容器内 CPU、内存、网络、磁盘、进程等实验场景

### **AHAS Service**

AHAS Chaos 的后端服务,用来下发演练命令,管理演练数据,主要的功能特点如下:

- 功能完善
  - 通过流程编排来组织故障演练场景。
  - 资源管理帮助您更方便的了解演练的机器资源使用情况。

### ● 稳定可靠

- 分布式的部署方式和完善的监控系统保证了演练功能的高可用性。
- 在服务不可用情况下,您可以通过手动执行命令来恢复演练。

# 4.产品优势

# 4.1. 功能优势

本文向您介绍故障演练产品的功能优势,故障演练以下简称 AHAS Chaos。

### 灵活的流程编排

AHAS Chaos 将故障演练的环节分为了准备、注入、检查以及恢复四个阶段,每个阶段除了系统初始化完成的必要节点之外,您也可以根据需要添加所需的流程节点。

● AHAS Chaos 支持一次演练包含多个定义的故障场景,同时您可以定制这些场景的运行方式,选择依次进行故障注入或同时注入多个场景,通过不同的策略配置来达到不同的故障注入效果。

### 丰富的故障场景

丰富的故障场景也是 AHAS Chaos 的一大特色,包括了以下场景:

- 常见的基础设施资源例如CPU、内存、磁盘等。
- 应用级别的故障注入,目前只支持 Java 应用,后续将陆续推出对于 NodeJs 和 C++ 的应用故障注入。
- 云原生领域的演练场景。

无论您是需要设置集群级别的大规模故障还是应用级别的请求级别细粒度故障,都可以在 AHAS Chaos 找到适合的场景,下图是 AHAS Chaos 提供的部分故障场景。



### 多样的专家经验

AHAS Chaos 将阿里内部多年的故障演练经验浓缩成了专家经验,专家经验具有以下优点:

- 专家经验都来自于阿里内部经常演练的场景,保证了演练场景的真实性以及实用性。
- 专家经验不但包括了可执行的演练流程,而且还描述了专家经验试图解决的问题以及针对的系统架构弱点。
- 专家经验极大的提升了演练创建的效率,您可以基于专家经验配置好的流程一键生成自己的演练。

### 安全的演练防护

故障演练·产品优势

在保护您的演练安全性上 AHAS Chaos 也做了非常多的防护措施。

- 在演练的任意一个环节,您都可以随时终止演练,每一个终止操作都会自动恢复注入的场景。
- 您可以一键终止所有正在运行当中的演练。
- 您可以配置演练自动的恢复时间,防止因演练时间过长而忘记恢复演练引发的不必要问题。
- 您可以通过全局恢复功能来配置自动恢复的策略,当某个指标符合某个要求时自动恢复演练。

### 深度集成的阿里云产品

AHAS Chaos 和阿里云的许多产品如 ARMS、SLS、EDAS、OTS 以及架构感知服务等做了深度集成,通过授权您可以实现以下功能。

- 对依赖的阿里云组件进行故障注入。
- 基于接入的阿里云监控系统数据如 ARMS 来丰富演练检查和恢复的手段。
- 通过 RAM 服务来授权不同账号的演练权限,提升演练的安全性。

# 4.2. 围绕混沌工程的平台实践

本文主要介绍AHAS Chaos是如何围绕混沌工程来打造故障演练服务,您可以了解到混沌工程的基本知识和 AHAS Chaos的优势。

### 混沌工程和故障演练

首先您需要了解混沌工程和故障演练的关系。以下是混沌工程官方定义:混沌工程是在分布式系统上进行实验的学科,目的是建立对系统抵御生产环境中失控条件的能力以及信心。因此混沌工程是一门学科,它提供了基本的理论指导,而故障演练是混沌工程的具体实践,通过向目标系统注入真实可能发生的故障来考量系统的稳定性。

### 混沌工程和AHAS Chaos

AHAS Chaos是以混沌工程为理论指导的故障演练平台,目标是成为混沌工程领域的最佳平台实践,因此在功能设计上紧紧围绕了混沌工程的基本原则。

### 原则1建立一个围绕稳定状态行为的假说

- 稳定状态规定了在演练开始前需要定义好观测的指标,这些指标需要可以真实地反应系统的健康程度,并且您还需要知道当前的故障场景对指标可能造成的影响。
- 假说就是在演练之前您需要知道演练可能会对系统有什么影响,并且通过演练去验证。

以上两点强调了系统指标的重要性,在实验期间您需要持续监控这些指标,并且根据指标做出相应的措施, 例如进一步扩大演练范围或者停止演练等。

AHAS Chaos提供了全局监控配置,可以帮助您在演练期间实时监测系统状态,及时处理预期之外的情况。 AHAS Chaos除了提供一些基础的系统指标如CPU、内存、网络、磁盘之外,还集成了阿里云的ARMS,如果您的应用已经接入ARMS就可以在AHAS Chaos里直接观测到对应的指标,后续也将会集成更多的监控系统,功能如下图所示。更多信息,请参见创建演练。



### 原则2多样化真实世界的事件

在故障演练开始之前,您需要思考以下问题,在现实生活中,系统已经发生过哪些问题?可能会发生什么样的新问题?有哪些问题是正在解决中的?这些问题可以帮助您在设计演练流程时选择和系统业务有关的更优场景。

为了帮助您选择合适的演练场景,AHAS Chaos在功能设计上做了以下两点:

● 丰富的演练场景。

首先在全面性上AHAS Chaos的演练场景覆盖了从基础设施、应用层到容器服务、云原生等多维度的领域,在场景真实性上AHAS Chaos也做了严格的筛选,无论是基础设施还是应用层针对的都是通用的组件,是被大多数的系统和业务所依赖的,例如CPU、磁盘、网络、MySQL等,此外AHAS Chaos还提供了阿里云组件的故障注入能力。

演练经验。

如果说演练场景是基于组件维度的,那么演练经验就向您提供了基于架构维度和故障效果维度的场景筛选,您无需手动配置流程,直接使用系统推荐的模板就可以生成演练,功能如下图所示。



### 原则3在生产环境中运行实验

混沌工程推荐故障演练是在生产环境中进行,主要的原因有以下两点:

- 系统的行为会根据环境和流量模式的变化,例如系统依赖的组件在测试环境和生产环境会有比较大的差异。
- 系统的监控和人员的应急响应在测试环境和生产环境也会有比较大的差异。

AHAS Chaos无法知道您当前的演练环境是否属于生产环境,为了更好地演练我们建议您将生产环境的演练和测试环境演练进行区分,您可以通过Chaos提供的Namespace功能将演练分为测试环境和线上环境,针对不同的环境指定响应的演练策略。



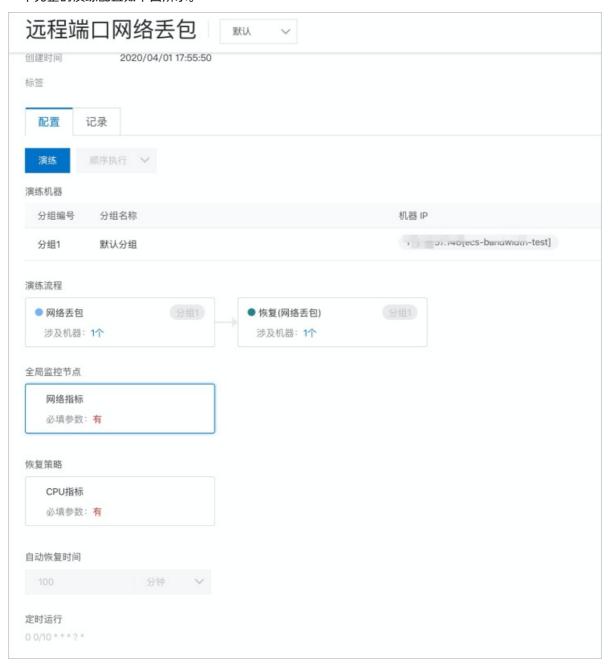
故障演练·产品优势 应用高可用服务

### 原则4持续自动化运行实验

由于系统、业务以及依赖的环境是在不断地变化,故障演练也需要持续地进行下去,因此自动化的故障演练是必不可少的环节。

演练平台必须高效安全,为此Chaos提供了以下能力:

- 提供了演练流程编排能力和多种演练运行方式。
- 可以根据您配置的恢复参数来自动恢复演练。
- 您还可以通过配置定时任务来自动化运行演练。
- 一个完整的演练配置如下图所示。



原则5最小化爆炸半径

最小化爆炸半径意味着混沌工程的影响范围必须得到控制,逐渐扩大故障范围,要保证演练是可控的,因此在演练当中需要您时刻关注在稳态假设中配置好的系统指标,如果影响范围超出了预期,请立刻终止演练,并且修复问题。如果影响范围没有达到预期,您也需要考虑下场景选择的合理性,演练的范围大小,或者其他的问题。

AHAS Chaos从多方面来控制演练的爆炸半径:

● 机器和场景选择方面。

AHAS Chaos已经将安装了演练探针的机器分成了主机和容器服务K8s(Kubernetes)集群两种维度,方便您控制机器范围,同时在演练场景上,AHAS Chaos针对主机和K8s可以选择的场景也做了区分,防止因为机器或者场景的错误选择而导致演练失败。

• 一键停止演练按钮。

AHAS Chaos提供了多个演练停止入口来让您立刻终止演练,并且演练终止都会自动恢复已经注入的故障。具体操作,请参见停止演练。

- 在演练列表页面,您可以一键停止所有的演练任务。
- 在演练执行界面, 您可以立刻终止演练。
- 在演练流程节点上,您也可以单击停止来终止演练。
- 基于指标自动恢复演练。

AHAS Chaos不但提供了多种手动演练入口,还可以配置自动恢复策略,请参见以下两种策略。

- 基于监控指标的自动恢复,如果指标触发了恢复规则,系统会自动执行演练恢复操作。
- 基于演练时长的自动恢复。

### 更多信息

关于AHAS Chaos的更多信息,请参见Chaos Engineering的历史、原则以及实践和混沌工程实践经验:如何让系统在生产环境中稳定可靠。

# 4.3. 开源ChaosBlade与商业化AHAS CHAOS故障演练对比

ChaosBlade是一款简单易用且功能强大的混沌实验实施工具,AHAS CHAOS故障演练是ChaosBlade的商业化产品。本文介绍AHAS CHAOS故障演练与开源ChaosBlade的能力对比。

### 产品简介

开源ChaosBlade与商业化AHAS CHAOS故障演练的特点如下:

- 开源ChaosBlade具备使用简单、场景丰富、演练模型标准化等一系列特点,可以帮助企业快速实施混沌工程。
- 商业化AHAS CHAOS故障演练是阿里巴巴内部广泛使用的演练平台云上版本,具有灵活的流程编排、丰富的故障场景等特点,可以帮助企业提升分布式系统的容错能力,保障在企业上云或往云原生系统迁移过程中业务的连续性。AHAS CHAOS故障演练在ChaosBlade的基础上,支持了更多的业务功能。

### 场景侧

AHAS CHAOS故障演练除了覆盖ChaosBlade已有的故障演练能力,同时具备了阿里云云产品和场景化的演练能力。

项目	AHAS CHAOS故障演练	ChaosBlade
CPU演练	支持	支持

项目	AHAS CHAOS故障演练	ChaosBlade
内存演练	支持	支持
网络演练	支持	支持
应用进程演练	支持	支持
K8s演练	支持	支持
阿里云云服务演练(ECS、Redis)	支持	不支持
场景化——微服务强弱依赖演练	支持	不支持
场景化——同城容灾演练	支持	不支持
场景化——消息服务演练	支持	不支持

② 说明 场景化的演练能力即针对某个业务组件或者特定场景的一整套演练实施方案,可以根据组件 部署架构来自动生成一批针对性的演练方案,大大降低了演练的实施难度。

# 平台侧 核心能力

项目	AHAS CHAOS故障演练	ChaosBlade
基础的故障注入、应用管理等功能	支持	支持
高可用的演练服务保障	支持	不支持
演练场景参数自动识别	支持	不支持
演练流程编排(串行、并行)	支持	不支持
演练流程安全防护	支持	不支持
演练专家经验库	支持	不支持
基于系统架构的组件化演练	支持	不支持
大规模的集群演练能力	支持	不支持

### 快速接入

项目	AHAS CHAOS故障演练	ChaosBlade
阿里云ECS快速接入	支持	不支持
K8s集群快速接入	支持	不支持
自建系统公网快速接入	支持	不支持

应用高可用服务 故障演练·产品优势

### 运维侧

项目	AHAS CHAOS故障演练	ChaosBlade
主子账号权限控制	支持	不支持
基于人员项目权限多维度的演练空间管理	支持	不支持

### 咨询侧

项目	AHAS CHAOS故障演练	ChaosBlade
阿里云演练专家答疑	支持	不支持

② 说明 您可以通过购买<mark>故障演练企业包</mark>并加入钉钉群(群号: 23196438)联系我们,开发人员会根据您不同的企业系统架构、业务特征来提供演练场景、演练方案、企业落地实施等演练的咨询服务,帮助您更快更好地落地故障演练,保障企业业务的连续性。

故障演练· 安装探针 应用高可用服务

# 5.安装探针

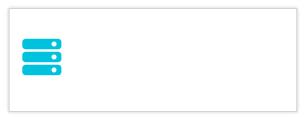
# 5.1. 安装探针概述

为您的主机等资源安装后,即可使用AHAS架构感知和故障演练功能,能直观地看到应用对基础架构的依赖 关系、组件间的依赖关系,以及对指定机器进行故障演练。应用高可用探针(即AHAS探针)

### 接入

AHAS架构感知和故障演练可以接入云服务器ECS、自建Kubernetes、容器服务K8s版和普通Linux主机的资源。根据主机所处环境不同,提供不同的接入方式。

⑦ 说明 Linux主机只支持在公网地域接入;云服务ECS只支持在非公网地域接入。



#### 接λFCS

架构感知和故障演练支持对阿里云云服务器ECS提供一键接入和 即 功能。

### 接入自建Kubernetes

架构感知和故障演练支持对自建的Kubernetes安装探针,您可以 关应用组件的依赖关系,并进行故障演练。

### 接入容器服务K8s版

接入成功后, 您可以执行以下操作:

• 架构感知: 查看云资源视图、应用视图、风险视图等。

● 故障演练:对应用进行故障演练。

### 5.2. 接入ECS

AHAS架构感知和故障演练支持在ECS上一键安装应用高可用探针。安装成功后,您可以直观地查看应用对基础架构的依赖关系,组件间的依赖关系,以及对应用进行故障演练。本文介绍如何快速接入ECS。

### 前提条件

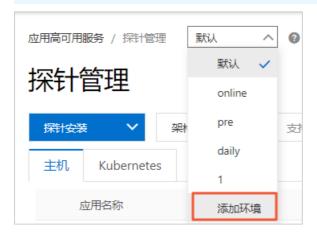
确保待安装探针的ECS系统为Linux或Windows。

### 操作步骤

1. 登录AHAS控制台,在顶部导航栏选择ECS所在地域。

○ 注意 已开通的地域包括华南1(深圳)、华北2(北京)、华东2(上海)、华北3(张家口)、华东1(杭州),其他地域的阿里云ECS主机可以通过公网地域,以Linux主机形式接入。具体操作,请参见接入主机。

- 2. 在左侧导航栏选择探针管理。
- 3. (可选)在探针管理页面上方的下拉列表中,选择添加环境,并在添加环境对话框中填写环境名称。
  - ② 说明 每个地域会有一个默认(Default)环境。您也可以添加自定义环境,如开发环境、测试环境等。不同环境的资源逻辑隔离。



- 4. 在探针管理页面,选择探针安装>安装故障演练探针。
- 5. 在安装探针页面,单击阿里云ECS。
- 6. 您可以选择以下任意一种方式安装探针:
  - 注意 Windows操作系统目前仅支持手动安装。
  - 自动安装:
    - a. 单击目标主机右侧操作列的单击安装。
    - b. 在**安装探针**对话框选择或新增要安装探针的应用。
    - c. 单击安装。
    - ② 说明 若为多台主机批量安装探针,请选中目标主机后,在页面左下角单击批量安装。
  - 手动安装:
    - a. 单击目标主机右侧操作列的手动安装。
    - b. 复制手动安装插件的命令。
    - c. 在本地执行安装插件的命令。
- 7. 单击下一步, 查看已安装的探针。

故障演练· 安装探针 应用高可用服务

8. 单击完成。

### 结果验证

创建完成后,您可以登录AHAS控制台,在左侧导航栏选择探针管理,在探针管理页面的主机页签,可查看到接入的应用以及探针信息。

# 5.3. 接入自建Kubernetes集群

AHAS架构感知和故障演练支持接入自建Kubernetes集群,可以自动侦测容器环境包含的ECS主机、容器组、容器、进程等。本文介绍如何将探针接入自建Kubernetes集群。

### 前提条件

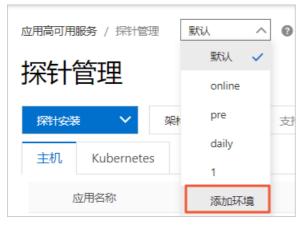
确保您的Kubernetes api-server组件接口版本在1.10及以上。

### 操作步骤

1. 登录AHAS控制台,在顶部导航栏选择集群所在地域。

### ? 说明

- 对于阿里云VPC环境的Kubernetes集群,已开通的地域包括华南1(深圳)、华北2(北京)、华东2(上海)、华北3(张家口)、华东1(杭州)。其他地域的用户,可以通过公网地域接入。
- 如果您的Kubernetes应用有公网连接,均可通过选择公网地域接入AHAS控制台。
- 2. 在左侧导航栏选择探针管理。
- 3. (可选)在探针管理页面上方的下拉列表中,选择添加环境,并在添加环境对话框中填写环境名称。
  - ② **说明** 每个地域会有一个默认(Default)环境。您也可以添加自定义环境,如开发环境、测试环境等。不同环境的资源逻辑隔离。



- 4. 在探针管理页面,选择探针安装 > 安装故障演练探针。
- 5. 在安装探针页面,单击自建Kubernetes。
- 6. 选择以下任意一种方式安装Helm Chart包。
  - 单击下载Helm Chart包。
  - 执行相应的命令下载。

公网

华东1(杭州)

华东2(上海)

华北2(北京)

华北3(张家口)

华南1 (深圳) 7. 安装高可用探针。

wget http://ahasoss-cn-public.oss-cn-hangzhou.aliyuncs.com/agent/prod/1.13.2/ahas1.13.2.tgz -O ahas-1.13.2.tgz

Helm v2安装探针:

0

■ 非公网:

helm install --set env.name=default --set controller.cluster\_id=<替换为集群ID, 取值无特殊要求> --set controller.cluster\_name=<替换为集群名称,取值无特殊要求> --namespace aha s --name ahas ahas-1.13.2.tgz

■ 公网:

helm install --set env.name=default --set license=<替换为真实license> --set controll er.cluster\_id=<替换为集群ID,取值无特殊要求> --set controller.cluster\_name=<替换为集群名称,取值无特殊要求> --namespace ahas --name ahas ahas-1.13.2.tgz

- Helm v3安装探针:
  - 非公网:

helm install ahas ahas-1.13.2.tgz --namespace ahas --set env.name=default,controlle r.cluster\_id=<替换为集群ID,取值无特殊要求>,controller.cluster\_name=<替换为集群名称,取值无特殊要求>

■ 公网:

helm install ahas ahas-1.13.2.tgz --namespace ahas --set env.name=default,license=< 替换为真实license>,controller.cluster\_id=<替换为集群id,取值无特殊要求>,controller.cluster name=<替换为集群名字,取值无特殊要求>

#### 执行成功后,返回如下:

8. 在安装探针页面单击下一步,并在查看数据页面检查已安装的探针信息。单击完成。

### 结果验证

故障演练·安装探针 应用高可用服务

创建完成后,您可以登录AHAS控制台,在左侧导航栏选择**探针管理**,在**探针管理**页面单击Kubernetes页签,可查看到接入的集群名称以及探针信息。

# 5.4. 接入容器服务K8s版

AHAS架构感知提供了针对容器服务K8s环境的可视化展示能力,自动侦测容器环境包含的ECS主机、容器组、容器、进程和云服务等组件,绘制组件之间的拓扑关系,持续记录,跟踪变化。AHAS的故障演练提供了对容器演练的功能。本文介绍如何将探针接入容器服务K8s版以及如何卸载该探针。

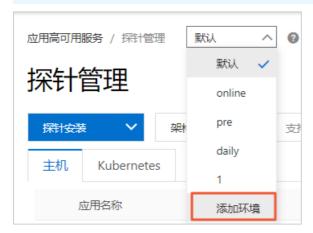
### 前提条件

快速创建Kubernetes托管版集群

### 接入步骤

在<mark>容器服务管理控制台</mark>安装AHAS组件后,即可自动侦测容器环境包含的ECS主机、容器组、容器、进程和云服务等组件。具体步骤如下:

- 1. 登录AHAS控制台,在顶部导航栏选择集群所在地域。
- 2. 在左侧导航栏选择探针管理。
- 3. (可选)在探针管理页面上方的下拉列表中,选择添加环境,并在添加环境对话框中填写环境名称。
  - ⑦ 说明 每个地域会有一个默认(Default)环境。您也可以添加自定义环境,如开发环境、测试环境等。不同环境的资源逻辑隔离。



- 4. 在探针管理页面,选择探针安装 > 安装故障演练探针。
- 5. 在安装探针页面,单击容器服务。
- 6. 在**应用目录-ack-ahas-pilot** 页面,单击参数页签,可查看安装该组件的默认参数值。如需修改,请参考以下参数说明。参数说明

参数	说明	默认值
controller.region_id	目标集群所在的地域,例如cn- hangzhou、cn-beijing、cn- shenzhen。	cn-hangzhou
resources.requests.cpu	AHAS Pilot占用的CPU。	0.05

参数	说明	默认值
resources.requests.memory	AHAS Pilot占用的内存。	200 Mi
resources.limits.cpu	AHAS Pilot占用的CPU最高限制, 例如,0.2。	0.2
resources.limits.memory	AHAS Pilot占用的内存最高限制, 例如,200 Mi。	200 Mi

7. 在应用目录-ack-ahas-pilot 页面右侧的创建面板中,选择集群,单击创建。

### 结果验证

创建完成后,您可以登录AHAS控制台,在左侧导航栏选择**探针管理**,在**探针管理**页面单击Kubernet es页签,可查看到接入的集群名称以及探针信息。

### 常见问题

如果探针安装没有成功,请按照以下方式排查:

● 检查是否选择了正确的地域(Region):

在AHAS控制台顶部菜单栏选择的地域,需要与安装AHAS Pilot 时参数 controller.region\_id 配置的地域一致。

查看参数 controller.region id 的步骤如下:

- i. 登录容器服务管理控制台, 在左侧导航栏单击集群。
- ii. 在集群列表页面中,单击目标集群名称或者目标集群右侧操作列下的详情。
- iii. 在集群管理页左侧导航栏中,选择应用 > Helm。
- iv. 找到发布名称为ahas的集群,单击操作列的详情。
- v. 单击参数页签, 查看地域参数 env. region 的值。
- 检查是否开通了AHAS:

登录<mark>开通AHAS服务</mark>页面验证。如果已开通,会提示跳转到AHAS控制台。

● 检查是否已授权AHAS服务:

登录授权AHAS服务页面验证。如未授权,单击同意授权。

### 卸载步骤

卸载容器服务中的探针(即AHAS应用高可用服务组件)的步骤如下:

- 1. 登录容器服务管理控制台, 在左侧导航栏单击集群。
- 2. 在集群列表页面中,单击目标集群名称或者目标集群右侧操作列下的详情。
- 3. 在集群管理页左侧导航栏中,选择应用 > Helm。
- 4. 找到发布名称为ahas的集群,单击其操作列的删除。
- 5. 在删除应用的对话框中,选中清除发布记录,单击确定。

### 5.5. 接入主机

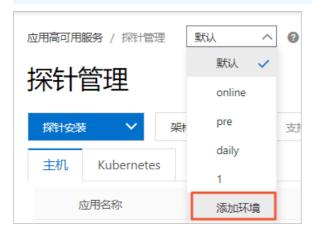
在普通主机上安装AHAS探针后,即可体验架构感知和故障演练功能,直观地查看应用对基础架构的依赖关系,组件间的依赖关系,以及对应用进行故障演练。本文介绍如何将探针接入主机,主机类型包括Linux和Windows。

故障演练·安装探针 应用高可用服务

### 操作步骤

- 1. 登录AHAS控制台,并在顶部菜单栏选择地域为公网。
- 2. 在左侧导航栏选择探针管理。
- 3. (可选)在探针管理页面上方的下拉列表中,选择添加环境,并在添加环境对话框中填写环境名称。

② 说明 每个地域会有一个默认(Default)环境。您也可以添加自定义环境,如开发环境、测试环境等。不同环境的资源逻辑隔离。



- 4. 在探针管理页面,选择探针安装 > 安装故障演练探针。
- 5. 在安装探针页面,单击主机。
- 6. 在安装探针页面,根据不同的操作系统 (Linux、Windows) 复制命令。
- 7. 登录主机,执行复制的命令。
  - ☐ 注意 Linux操作系统请使用Root用户执行复制的命令。
- 8. 单击下一步,并在查看数据页面查看安装的探针信息,然后单击完成。

### 结果验证

创建完成后,您可以登录AHAS控制台,在左侧导航栏选择探针管理,在探针管理页面的主机页签,可查看到接入的应用以及探针信息。

# 6.开始演练

# 6.1. 演练概述

基于阿里巴巴多年业务的真实线上故障库的积累,AHAS故障演练模块为您预定义了丰富的测试任务,检验应用的高可用能力。故障演练支持对部署在阿里云ECS实例、容器服务Kubernetes集群、K8s架构类应用、lava类应用进行故障演练。

- 一次完整的故障演练包括以下四个阶段:
- 安装探针:对指定机器进行演练,需要在机器上面安装故障演练探针,探针的作用是下发故障演练执行命令。
- 创建演练:配置演练基本信息、演练对象和演练全局参数。可同时选择多个故障类型。
- 执行演练:将故障注入机器,可通过演练时需曲线、演练参数、演练日志等检查故障注入的效果是否符合 预期。
- 停止阶段:清除故障。当故障演练自动结束、您主动终止或者演练中的任何环节出现异常后,系统都会进入恢复阶段,自动清除相应的故障,使故障演练对象恢复演练前的状态。



### 6.2. 接入演练应用

故障演练支持接入容器服务Kubernetes应用以及Java、GO、PHP等语言的应用,本文介绍如何在故障演练中接入新应用。

### 接入Java应用

通过配置JVM启动参数来指定应用和应用分组,用于精确划分该机器所归属的应用或应用分组(与安装探针时指定的应用或应用分组不冲突)。

1. 配置JVM启动参数。

AppName只能包含字母、数字和特殊字符下划线 ( ) 和短划线 (-)。

-Dproject.name=应用名 -Dproject.group.name=应用分组名

请根据实际情况替换以上的值,上述配置默认值如下:

- o project.name : 默认值 ahas-default-app 。
- o project.group.name : 默认值 ahas-default-app-group 。

○ 注意 在已部署故障演练探针的机器上,修改JVM启动参数并重启,应用会自动识别并生效, 无需重新部署故障演练探针。

- 2. 启动应用。
- 3. 登录AHAS控制台, 在左侧导航栏选择**故障演练 > 概览**。
- 4. 在左侧导航栏单击探针管理,然后探针管理单击右上角的接入探针。
- 5. 在选择环境页面单击阿里云ECS。
- 6. 在安装应用高可用插件页面安装探针,在目标主机右侧操作列单击单击安装。
- 7. 填写应用、应用分组信息。
  - 已有应用,则选择应用名称与应用分组,单击安装。

- 新增应用,则填写应用名称与应用分组,单击安装。
- 8. 单击下一步, 查看已安装的探针。
- 9. 单击完成。

### 接入Kubernetes容器化应用

通过Pod标签识别其所归属的应用或应用分组,配置如下。

1. 配置Pod标签。

标签的值只能包含字母、数字和特殊字符下划线(\_)和短划线(-)。 在模板(YAML格式)中将以下lables配置添加到spec > template > labels层级下:

labels:

ahas.aliyun/app-instance: 应用名 ahas.aliyun/app-group: 应用分组名

根据实际情况替换以上的值,如果不配置以上的值,会再次识别是否包含 app-group-name (容器服务应用配置)、 edas.oam.acname (EDAS应用配置)、 app 、 k8s-app 标签配置作为应用名,应用分组名格式默认为:应用名-group。

根据标签识别应用,标签优先级如下: ahas.aliyun/app-instance > app-group-name > edas.oam.acname > app > k8s-app。

☐ 注意 在已部署故障演练探针的集群,修改Pod标签即可生效,无需重新部署故障演练探针。

- 2. 登录AHAS控制台, 在左侧导航栏选择故障演练 > 概览。
- 3. 在左侧导航栏单击探针管理,然后探针管理单击右上角的接入探针。
- 4. 在选择环境页面单击容器服务,创建ack-ahas-pilot。具体操作,请参见架构感知监控。

### 接入其他应用

通过主机部署,接入其他应用操作步骤如下。

- 1. 登录AHAS控制台, 在左侧导航栏选择故障演练 > 概览。
- 2. 在左侧导航栏单击探针管理, 然后探针管理单击右上角的接入探针。
- 3. 在选择环境页面单击阿里云ECS。
- 4. 在安装应用高可用插件页面安装探针,在目标主机右侧操作列单击单击安装。
- 5. 填写应用、应用分组信息。
  - 已有应用,则选择应用名称与应用分组,单击**安装**。
  - 新增应用,则填写应用名称与应用分组,单击安装。
- 6. 单击下一步, 查看已安装的探针。
- 7. 单击完成。

### 6.3. 安装故障演练探针

对指定机器进行演练时,需要在机器上面安装故障演练探针,探针的作用是下发故障演练执行命令。

### 背景信息

一次完整的故障演练包括以下四个阶段:

安装探针 > 创建演练 > 执行演练 > 停止演练

### 操作步骤

- 1. 登录AHAS控制台。
- 2. 在左侧导航栏选择探针管理,在探针管理页面,选择探针安装 > 安装故障演练探针。
- 3. 在安装探针页面选择安装探针的环境,具体步骤,请参见安装探针概述。

### 后续步骤

- 创建演练
- 执行演练
- 停止演练

### 6.4. 配置RAM权限

AHAS支持对云服务器ECS(Elastic Compute Service)和容器服务ACK(Alibaba Cloud Container Service for Kubernetes)进行演练,为了控制被演练对象的范围,AHAS故障演练支持对RAM子账号进行授权配置。

### 配置方法

- 1. 登录RAM访问控制。
- 2. 在左侧导航栏中选择权限管理 > 权限策略。
- 3. 在权限策略页面,单击创建权限策略。
- 4. 在**创建权限策略**页面单击**脚本编辑**页签,填写脚本,然后单击下一步。 以如下配置为例进行说明。

- Action: 配置为ahas:\*拥有资源的全部权限。
  - 创建演练权限: 配置为 ahas:CreateExperiment , 授予该权限后,可以对资源进行演练配置操作 (含更新操作)。
  - 执行演练权限: 配置为 ahas:ExecuteExperiment , 授予该权限后, 可对资源进行演练执行操作 (演练可执行的条件为 "子账号拥有该演练中所有资源的执行权限")。
- Resource: 目前仅支持ECS和ACK两种资源,请根据实际情况配置允许演练的资源,具体语法请参见创建自定义权限策略说明。
- 5. 在基本信息区域填写策略名称,然后单击确定。
- 6. 授权新创建的权限策略给用户,详情请参见为RAM用户授权。

故障演练·开始演练

### 应用权限配置方法

? 说明 应用及应用分组名称不可包含中文。

应用对应的权限配置方法请参见以下操作步骤。

- 1. 登录RAM访问控制。
- 2. 在左侧导航栏中选择权限管理 > 权限策略。
- 3. 在权限策略页面,单击创建权限策略。
- 4. 在创建权限策略页面,单击脚本编辑页签,填写脚本,然后单击下一步。

- Action: 配置为ahas:\*拥有资源的全部权限。
  - 创建演练权限: 配置为 ahas:CreateExperiment , 授予该权限后,可以对资源进行演练配置操作 (含更新操作)。
  - 执行演练权限:配置为 ahas:ExecuteExperiment ,授予该权限后,可对资源进行演练执行操作 (演练可执行的条件为"子账号拥有该演练中所有资源的执行权限")。
- Resource: 支持按应用和应用分组进行资源划分,以不同维度配置授权。
- 。 Resource表达式格式:

```
acs:ahas:region:accountId:environment/appInstance/appGroup/ip
```

- Resource表达式说明:
  - accountId: 用户ID。
  - environment:环境。指AHAS平台上的环境,取值是环境名,例如默认的default。
  - applnst ance:应用。用户部署故障演练探针时确定,如果用户部署时不指定,会归属为默认应用(ahas-def ault-app)。
  - appGroup:应用分组。用户部署故障演练探针时确定,如果用户部署时不指定,会归属为默认应用分组(ahas-default-app-group)。
  - ip: 资源IP, 如ecs ip, pod ip或node ip等。
- 5. 在基本信息区域填写策略名称,然后单击确定。

### 示例

示例1

如上示例1所示,持有该权限的人对**online环境 > business应用 > business\_host分组**内所有资源拥有配置及更新权限。

#### 示例2

如上示例2所示,持有该权限的人对default 环境 > order应用 > order\_1分组 > 10.1.0.1设备拥有演练执行权限。

# 6.5. 创建演练

创建演练包括配置演练基本信息、演练对象和演练全局参数。

### 背景信息

一次完整的故障演练包括以下四个阶段:

安装探针 > 创建演练 > 执行演练 > 停止演练
---------------------------

### 操作步骤

- 1. 登录AHAS控制台, 在左侧导航栏中选择故障演练 > 我的空间。
- 2. 在我的空间页面,单击新建演练。
- 选择新建空白演练或从经验库新建。
   选择从经验库新建会从经验库里直接生成演练的配置基本信息。
- 4. 单击新建空白演练后,在演练配置页面,填写演练名称、演练描述、演练标签和演练空间。

配置项	配置说明
演练名称	填写演练名称。

配置项	配置说明
演练描述	为该演练添加描述,包括演练原因、演练场景以及演练可能造成的影响等信息。
演练标签	自定义演练标签,便于演练的查询和统计:
演练空间	填写需要创建演练的空间,便于管理成员和演练。

### 5. 配置演练对象。

i. 在演练对象区域设置分组名称,选择资源类型。

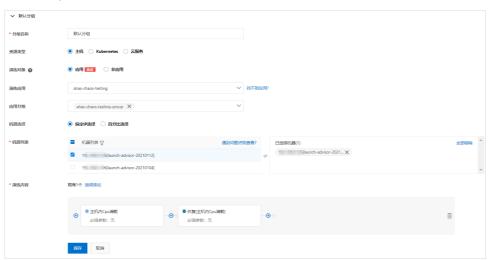
资源类型包括以下类型:

- Kubernetes类型:包括Kubernetes类型的应用。
- 主机类型:包括Linux系统资源和Windows系统资源。
- 云服务类型:包括ECS、RDS等云服务。
- ii. 选择演练对象。
  - 若选择演练对象为应用,则需要选择演练应用、应用分组、机器列表等。
  - 若选择**演练对象**为非应用,则需要选择机器列表。

### ? 说明

- 机器列表中可以选择一台或多台机器。
- 非应用的机器列表中可切换Linux或Windows机器类型进行查看。
- iii. 若选择演练对象为应用,则需要选择机器选择的方式。
  - 指定IP选择:选中机器列表中的机器即可,精确指定机器执行故障注入。
  - 百分比选择: 当演练对象为应用时,可使用**百分比选择**方式。滑动指针,设置1~100的百分比数值,每次执行演练时,系统会随机从所选应用分组下的存活机器中,按百分比选取目标机器,并执行故障注入。
- iv. 单击**添加演练内容**,然后按需选择演练场景。故障演练支持的场景,请参见<mark>演练场景说明</mark>。

v. 单击保存, 然后单击下一步。



- 6. 配置全局配置。在全局配置区域完成以下配置。
  - 以下配置以脚本方式制造CPU满载的演练场景为例。
    - i. 选择演练流程为顺序执行。
    - ii. 单击监控策略的**新增策略**,选择需要监控的指标,然后单击**确定**。在指标的面板中设置相关参数。
    - iii. 单击恢复策略的**新增策略**,选择需要恢复的指标,然后单击**确定**。
  - iv. 在指标面板中, 配置恢复规则和恢复策略。

例如配置恢复策略为: 当70%的机器满足system.cpu.util(综合利用率)等于100%且持续30s时,即可解除CPU满载,恢复初始CPU使用率。



- v. 设置自动恢复时间。
- 7. 单击下一步,在成功弹窗单击演练详情进入演练详情页,或单击返回首页返回概览页面。

### 后续步骤

故障演练·开始演练

演练创建成功后, 您可以:

- 执行演练 在演练列表中的某个演练的操作列,单击演练,执行演练。请参见执行演练。
- 查看并编辑演练详情 在演练列表,单击某个演练名称,可查看演练内容。单击页面右上角**编辑演练**,可修改演练内容。
- 拷贝演练 在演练列表中的某个演练的操作列,单击**拷贝**,拷贝一个同名的演练,您可以在此基础上编辑演练内容。
- 刪除演练 在演练列表中的某个演练的操作列,单击删除。

# 6.6. 执行演练

在执行故障演练过程中,您可以实时查看演练进度、每个演练活动的运行状态及结果,同时也能够随时结束 演练,进行恢复阶段的活动,清除故障演练影响。

### 背景信息

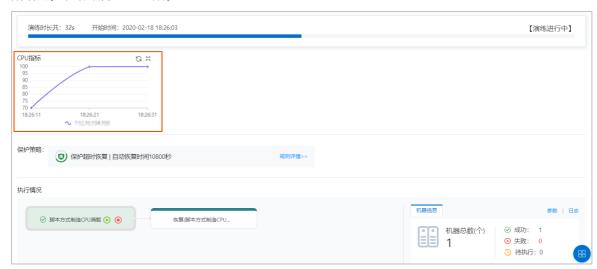
一次完整的故障演练包括以下四个阶段:



### 操作步骤

演练任务创建完成后,您可以直接执行演练。

- 1. 在左侧导航栏中选择故障演练 > 我的空间。
- 2. 在演练列表中单击目标演练任务右侧操作栏的演练,然后在开始执行演练?对话框中单击确认。 在演练名称列单击目标演练名称,在该演练详情页可以看出故障开始注入之后,目标机器的CPU指标开始增加,说明故障已经生效。



您可以随时关注演练进度、演练时长、活动运行的结果等。演练执行界面分为以下几个区域。

参数	描述
基本信息区域	包括了演练进度以及开始时间等信息。

参数	描述
指标展示区域	如果在演练参数设置里面配置了全局监控节点,则可以看到实时的系统指标数据。  • 指标数据将定时更新,您可以单击右上角的刷新图标手动请求数据。  • 如果演练尚未结束,时间范围为演练开始到当前时间,如果演练已经结束,时间范围为演练开始到演练结束时间。指标采集存在约1分钟延迟。  • K8s类场景中,用户配置的机器地址为Master地址,非故障实际生效的节点IP,此处展示的IP地址是实际生效的节点IP地址。例如用户配置了192.168.1.1,此为故障下发地址,实际影响的是K8s集群上192.168.1.2,192.168.1.3两个节点地址,那么监控数据会展示192.168.1.2,192.168.1.3。
保护策略区域	若配置了保护策略,则可以看到正在运行的保护策略列表。     演练执行后,保护策略开始执行。     若您手动终止演练,那么保护策略也会终止。
流程执行区域	流程执行区域展示了当前演练的节点运行情况以及当 前节点每台机器的执行情况。
节点操作说明	根据节点的通用配置,在节点上会有以下操作。  ②: 说明当前节点已经运行完毕,需要用户手动单击才能进入到下一个节点。  ③: 说明当前节点已经运行完毕,如果您无需再运行下一个节点,则单击终止整个演练任务。  ③: 说明当前节点运行失败,单击重试。
节点详情	单击任何一个节点,会在右侧展示节点的详情,包含以下信息。      机器信息:可以看到每一台机器的运行情况,如果当前机器执行错误,可以单击机器IP来查看具体的错误信息。      参数:可以看到演练节点的配置参数。      日志:可以看到演练运行过程中当前节点的执行日志。

### 后续步骤

停止演练

### 常见问题

故障演练常见问题

故障演练·开始演练

# 6.7. 停止演练

除了设置故障演练自动结束时间外,还可以手动停止演练。停止演练后,系统会进入恢复阶段,自动清除相应的故障,使故障演练对象恢复演练前的状态。

### 背景信息

一次完整的故障演练包括以下四个阶段:



### 自动停止

在创建演练过程中需设置自动恢复时间和保护策略,起到自动停止演练的作用。具体步骤,请参见创建演 练。

- 当演练时长超过设置的自动恢复时间,则会终止演练任务。
- 若为演练任务配置了保护策略,触发保护策略则会终止演练任务。

### 手动停止

您可以通过以下三种方式在AHAS控制台手动停止演练。

- 在左侧导航栏选择**故障演练 > 我的空间**,然后在演练列表中单击目标演练任务右侧操作列的停止,在停止演练的对话框中单击确定。
- 在演练详情页右上角单击终止,然后在停止演练的对话框中单击确定。
- 在演练详情页执行情况区域执行节点后单击终止。



# 6.8. 接入ARMS监控

故障演练时通过接入应用实时监控服务ARMS(Application Real-Time Monitoring Service)可以对演练过程的指标进行监控,包含JVM内存、JVM线程数、JVMGC相关、网络出入口流量、磁盘、CPU等指标。本文介绍如何在故障演练中接入ARMS并监控演练指标的操作步骤。

### 前提条件

已分别在故障演练和ARMS中接入应用,且两个应用的名称要保持一致。接入应用的操作,请参见安装探针概述和创建应用监控任务。

### 操作步骤

- 1. 登录AHAS控制台, 在左侧导航栏选择故障演练 > 我的空间。
- 2. 单击新建演练,选择新建空白演练,进入演练配置页面。
- 3. 单击新建空白演练后,在演练配置页面,填写演练名称、演练描述、演练标签和演练空间。

配置项	配置说明
演练名称	填写演练名称。

配置项	配置说明
演练描述	为该演练添加描述,包括演练原因、演练场景以及演练可能造成的影响等信息。
演练标签	自定义演练标签,便于演练的查询和统计:      输入标签内容时,系统会显示已有的标签。      一个演练场景中,最多可输入五个标签。      标签将自动保存,下次可通过搜索或直接从下拉列表中选择已有的标签。
演练空间	填写需要创建演练的空间,便于管理成员和演练。

4. 在配置页签,设置演练对象的参数。



- i. 设置分组名称。
- ii. 选择演练对象为应用。
- iii. 选择演练应用、应用分组和机器列表等。
  - ? 说明 机器列表中可以选择一台或多台机器。
- iv. 单击添加演练内容,然后按需选择演练场景。故障演练支持的场景,请参见演练场景说明。
- v. 单击**保存**,可添加多个演练分组。然后单击下一步。
- 5. 配置全局配置。
  - i. 选择演练流程。
  - ii. 在**监控策略**区域单击新增策略,在新增策略面板中选择需要监控的节点,并选择参数对应的指标,如查询JVM内存信息(ARMS),可选择新生代、老年代等相关指标。然后单击确定
  - iii. 在恢复策略区域单击新增策略,选择指标。
  - iv. 在**新增策略**面板中,选择恢复指标,然后单击**确定**。
  - v. 设置自动恢复时间和定时运行时间。
- 6. 单击下一步,在成功弹窗单击演练详情进入演练详情页,或单击返回首页返回概览页面。

#### 执行结果

在演练详情页,单击演练,在开始执行演练?对话框中单击确认执行演练后,可以观察监控的指标。



# 7.可视化演练

## 7.1. 什么是可视化演练

AHAS的可视化演练是基于架构视图来控制爆炸半径,并通过可视化的交付方式,直接面向演练效果发起的演练。通过可视化演练,您可以从可视化的系统架构来观察演练的爆炸半径,从而降低了您配置参数、控制爆炸半径的成本。

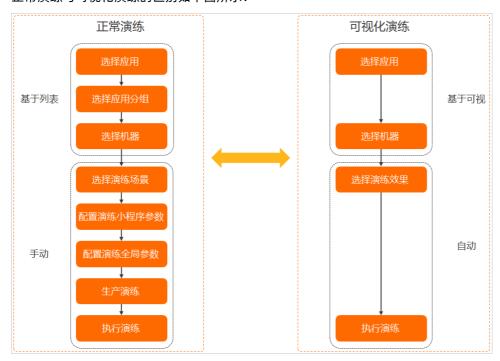
#### 可视化演练特性

AHAS Chaos的可视化演练具备以下几个特点:

- 架构可视化
  - 全局视角 通过全局视角,您可以看到整个业务系统的全貌,各个应用的演练状态以及彼此间的依赖关系。
  - 应用视角 您可以看到该应用的上下游依赖、应用所有的实例以及各个实例的演练任务细节。
- 面向效果快速演练
  - 基于架构可视化,您可以快速地选取需要发起演练的机器。
  - 您可以根据演练效果直接发起对应的演练,而不用根据自己想要的演练效果去配置相应的演练场景。
  - 演练具体的配置参数会自动生成,不再需要您手动填写。

#### 正常演练与可视化演练的区别

正常演练与可视化演练的区别如下图所示:



## 7.2. 使用可视化演练

本文介绍可视化演练展示信息以及如何快速发起可视化演练。

#### 全局架构视图

可视化演练的全局架构视图展示应用的机器数量、演练数量以及应用之间的调用关系等。

- 1. 登录AHAS控制台,在左侧导航栏选择故障演练 > 概览。
- 2. 在故障演练页面的左侧导航栏选择可视化演练。

界面展示可视化演练的全局架构视图。



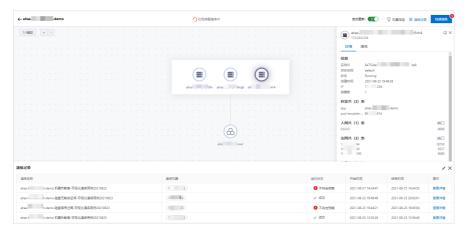
- 左侧显示所有应用列表。列出了应用的机器数量和演练数量,并支持按名称搜索。 您可通过应用名称来搜索指定应用。
- 右侧显示系统整体的架构。展示了应用之间的调用关系。
  - 您可在右侧架构中单击某个应用,展示应用详情视图。
  - 在页面右上角,您可单击**应用筛选**,选择需要展示的应用。
- o 演练状态观测。当有演练进行时,会将演练中的应用进行标识,让您直观地观察爆炸半径。



#### 应用详情视图

可视化演练中的应用详情展示应用的上下游服务依赖、所有机器实例以及演练任务记录等信息。

- 1. 登录AHAS控制台, 在左侧导航栏选择故障演练 > 概览。
- 2. 在故障演练页面的左侧导航栏选择**可视化演练**,并在**应用视图**界面右侧单击待查看应用的图标,进入应用详情页面。



- 中间区域展示了应用的上下游服务依赖以及所有机器实例。若需要查看某个实例的详情,可单击待查看的实例,右侧弹出实例的详情页面,展示实例的机器详情以及演练任务记录。
- 单击页面右上角的**演练记录**,可在下方区域查看该应用下的演练任务记录,包括演练名称、演练机器、运行状态、开始时间和结束时间等信息。 若需要查看某个演练的详情,可单击操作列的**查看详情**进行查看。

#### 快速发起演练

可视化演练支持根据演练效果快速的发起演练。

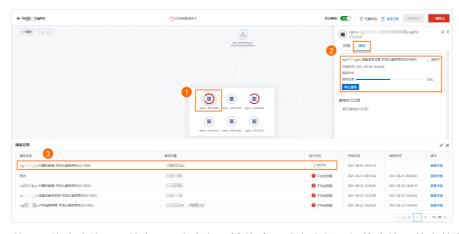
- 1. 登录AHAS控制台,在左侧导航栏选择故障演练 > 概览。
- 2. 在左侧导航栏选择可视化演练,并在应用视图界面选择待执行演练的应用,进入应用详情页面。
- 3. 在应用详情页面选择待执行演练的机器实例,右侧弹出实例的详情页面
- 4. 单击页面右上角快速演练,在右侧快速演练面板中设置对应参数。
  - i. 选择演练对象。

ii. 单击下一步,设置演练内容。



- a. 选择演练场景。
  - 可根据需要选择环境类的故障演练或是服务依赖类的故障演练。
  - 运行环境可选择机器负载高、内存消耗高、磁盘无剩余空间或磁盘读写过高场景。
  - 服务依赖 可选择本地服务丢失、本地服务延迟、调用下游服务丢失或调用下游服务延迟场景。
- b. 设置自动恢复时间。
- iii. 单击执行演练。

当演练进行时,您可在图示中三个区域内观测到正在执行的演练任务。



若需要终止演练,可单击页面右上角一键终止,或在右侧面板的演练页签中单击停止演练。

# 8.演练场景

## 8.1. 演练场景说明

故障演练场景是演练任务的核心。AHAS提供基础资源类场景和Kubernetes类场景,帮助分布式系统提升容错性和可恢复性。

每一个执行阶段的演练场景都对应一个恢复阶段的演练任务。恢复阶段的演练任务目的是清除故障演练的影响,使应用或服务恢复正常,通常不需要配置参数。本文不再介绍恢复阶段的演练活动。

#### 基础资源类场景

场景名称	特性
Node演练场景	Kubernetes集群中Node资源故障场景,目前包含基础资源中的CPU、网络和进程。每个Node场景下都包含通用的Node筛选参数,用于查找目标Node。
Pod演练场景	Kubernetes集群中Pod资源故障场景,包含删除Pod和Pod网络异常场景。每个Pod场景下都包含通用的Pod筛选参数,用于查找目标Pod。
Container演练场景	Kubernetes集群中Pod资源下的容器故障场景,目前包含删除容器以及容器内故障场景。每个容器故障场景下都包含通用的容器筛选参数,用于查找目标容器。

#### Java

#### K8s类场景

### 8.2. 基础资源类场景

### 8.2.1. CPU类场景

CPU类场景可以指定CPU使用率,也可以指定CPU满载的使用核数。帮助您在CPU在特定负载的环境下,验证应用或服务的性能指标以及监控告警、流量调度、弹性伸缩等能力。

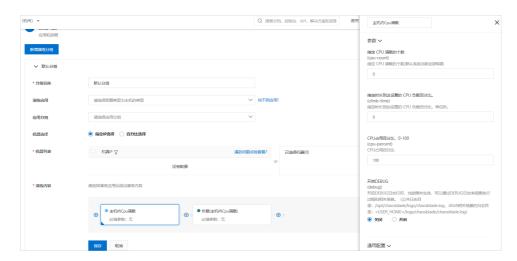
#### 参数说明

参数名称	参数说明
指定CPU满载的个数	指定使用率满载的CPU核数,可取值区间为 [0,应用总核数],默认值为0,表示全部核满载。
指定时长到达设置的 CPU 负载百分比	指定时长到达设置的CPU负载百分比,单位秒。
CPU占用百分比	指定CPU使用率,取值区间为 [0,100]。填写时不用填写百分号,例如填写80,则指定CPU的整体使用率达到80%。

#### 示例

指定全部核数CPU满载的演练场景配置如下:

故障演练· 演练场景 应用高可用服务



### 8.2.2. 网络类场景

网络类场景包含网络延迟、网络丢包和篡改域名解析等场景。帮助您在网络异常的情况下验证应用或服务的 容错能力。

#### 网络延迟

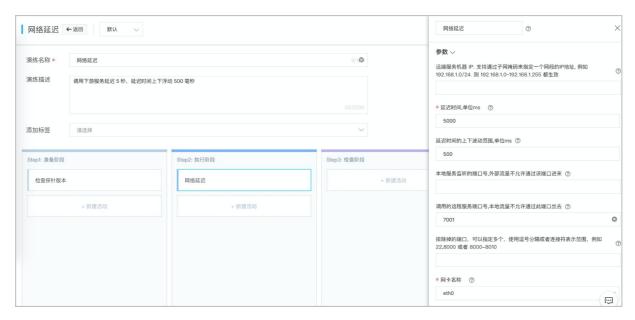
网络延迟场景可以指定网络延迟因素(例如网卡、本地端口、远程端口、目标IP等)和延迟时间,对应用或服务注入网络调用延迟故障。验证网络延迟情况下系统的容错能力。

#### 参数说明

参数名称	参数说明
网卡名称	具体的网卡设备,为必选项,例如eth0。
本地服务端口	本地服务监听的端口,外部流量不允许通过此端口进来。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000-8080。
远端服务端口	调用远程服务的端口,本地流量不允许通过此端口出去。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000-8080。
排除端口	无需注入网络延迟调用故障的端口,与本地服务端口和远程服务端口功能互 斥。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000- 8080。
远端服务IP	指定演练对象访问的远端服务IP。可以通过子网掩码来指定一个网段的IP地址,例如若填写192.168.1.0/24.,则192.168.1.0~192.168.1.255都生效。
延迟时间	指定网络延迟的时间,单位是毫秒,必填项。
延迟上下浮动时间	网络延迟时间的上下浮动范围,单位是毫秒。例如配置100,则表示所配置的延迟时间上下浮动100毫秒。

#### 示例:

应用A调用下游服务(服务端口是7001)延迟5秒,延迟时间上下浮动500毫秒,网络调用的网卡是eth0。演练场景配置如下:



#### 网络丢包

网络丢包场景是指通过指定网络丢包因素(例如网卡、本地端口、远程端口、目标IP等)和丢包百分比,对应用或服务注入网络丢包故障。验证网络丢包情况下系统的容错能力。

#### 参数说明

参数名称	参数说明
网卡名称	具体的网卡设备,为必选项,例如eth0。
本地服务端口	本地服务监听的端口,外部流量不允许通过此端口进来。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000-8080。
远端服务端口	调用远程服务的端口,本地流量不允许通过此端口出去。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000-8080。
排除端口	无需注入网络延迟调用故障的端口,与本地服务端口和远程服务端口功能互 斥。可以指定多个,使用逗号分隔,使用连接符表示范围。例如80,8000- 8080。
远端服务IP	指定演练对象访问的远端服务IP。可以通过子网掩码来指定一个网段的IP地址,例如若填写192.168.1.0/24.,则192.168.1.0~192.168.1.255都生效。
丢包百分比	网络调用丢包率,为必填项。取值为正整数,取值区间为[0,100],填写时不加百分号。

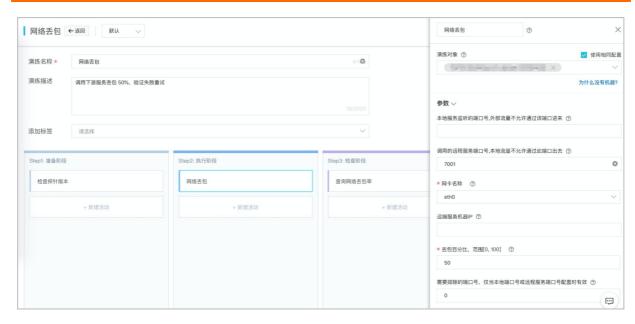
### ? 说明

- 排除端口不能与本地服务端口、远端服务端口同时使用。
- 若只配置网络名称,不配置本地服务端口、远端服务端口或排除端口,则执行演练时会影响该网 卡全部端口。

#### 示例:

应用B调用下游服务丢包率为50%,调用端口为7001,网络调用的网卡是eth0。则演练场景配置如下:

故障演练· 演练场景 应用高可用服务



#### 篡改域名解析

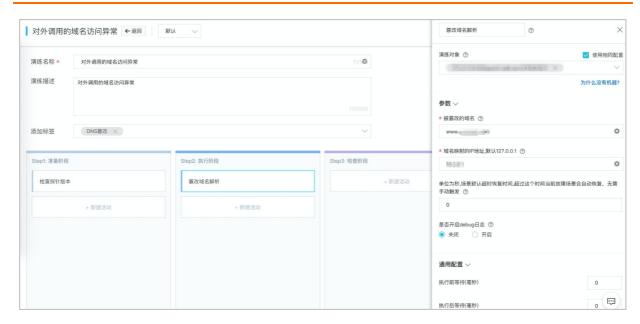
篡改域名解析场景是指通过修改目标主机的 host s文件,篡改域名地址映射,使域名访问异常。用于验证域名解析错误的情况下,应用或服务的容错能力。

#### 参数说明

参数名称	参数说明		
	指定需篡改的域名,必填项,例如www.aliyun.com。		
被篡改的域名	⑦ 说明 域名不添加http://或https://,并且域名后面不添加端口号。		
域名映射的IP	指定域名映射的IP,必填项,例如10.0.0.1。		

#### 示例:

为应用A配置篡改域名解析演练场景,示例如下:



## 8.3. Java 类

### 8.3.1. 虚拟机场景

本文列出了虚拟机常见的故障演练场景。

#### JvmCodeCache满

制造JVM CodeCache区域满的故障,CodeCache区域满会直接导致JIT编译关闭,从而引起业务系统性能下降。一般用于验证业务系统在高并发且系统性能下降的情况下,是否能够通过限流、快速扩容等手段保证业务连续性。

#### 参数说明如下:

参数名称	是否必选	默认值	参数说明
进程ID		无	Java进程的ID。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
开启Debug	否	否	选择是否开启Debug日志,用于排查演练 执行过程中遇到的问题。开启Debug后, 请 到~/logs/chaosblade/chaosblade.log 路径下查看日志。

#### Java产生OutOfMemoryError异常

填充JVM指定的内存区域,导致指定内存区域空间不足,引发OOM异常。

参数说明如下:

参数名称	是否必选	默认值	参数说明
内存区域	是	无	指定填充的JVM内存区域,可选项: 新生代、老生代(MetaSpace)、堆 外内存。
暴力模式	否	False	是否开启暴力模式,如果是暴力模式,在OOM发生之后也不会释放之前创建的内存,可能会引起应用进程无响应。
时间间隔	否	无	两次OOM异常间的时间间隔,只有在非暴力模式才生效,可以减缓GC的频率,避免因频繁的FullGC导致进程无响应。
数据块大小(MB)	否	无	每次填充内存的数据块大小,仅当内 存区域选择新生代或堆外内存时生 效。
进程ID		无	Java进程的ID。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该 关键字查找到唯一进程,使用 ps - ef   grep <key> 来尝试查找进 程,能找到唯一进程则正确。</key>
开启Debug	否	否	选择是否开启Debug日志,用于排查 演练执行过程中遇到的问题。开启 Debug后,请到 ~/logs/chaosblade/chaosblade.l og路径下查看日志。

### Java应用内部制造CPU满载

Java应用进程内制造CPU满载,表现为Java应用本身原因导致CPU高负载。

参数名称	是否必选	默认值	参数说明
指定CPU满载的个数	否	无	指定CPU满载的个数,默认系统当前全部核数。
进程ID		无	Java进程的ID。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
开启Debug	否	否	选择是否开启Debug日志,用于排查演练执行过程中遇到的问题。开启Debug后,请到~/logs/chaosblade/chaosblade.log路径下查看日志。

## 8.3.2. 代码逻辑场景

本文列出了故障演练支持的代码逻辑场景。

### 篡改Java方法返回值

修改Java指定方法的返回值,返回指定的值。

参数名称	是否必选	默认值	参数说明
类名	是	无	完整的类名,包含包名。例如:com.alibaba.service.XxxService。如果模拟接口故障,需填写接口的实现类。
方法名	是	无	方法名,例如: 方法getUser(Long userId),则填写getUser。如果存在多个重载方法,例如: getUser(String name) 和 getUser(Long userId) ,则对两个方法均生效。
返回值	是	无	指定方法的期望返回值,仅支持基础类型、基础类型的包装类、String类型。如果返回空值,填写null。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
进程ID		无	进程的ID。
受影响的请求数	否	0	限制最多发生故障的请求总数,每生效一次故障计数加1,累计发生故障请求数超出设定值后,请求则不再发生故障。填写数值小于等于0时,则表示不限制。
			限制发生故障的请求数占所有应该发生故障请求数的百分比,也可代表每次请求发生故障的概率。填写小于或等于0,则表示100%发生故障。
(%)	的请求占比 否 0	0	② 说明 仅填写百分比数字部分即可,即 80%,填写80。
请求过滤规则	否	无	通过脚本方式自定义规则,通过自定义规则决策是 否对请求产生故障。自定义规则生效前提为需满足 其它设定条件。
过滤规则执行阶段	否	无	自定义过滤规则执行的阶段,可选择Java方法调用 前执行或Java方法调用后执行。
开启Debug	否	False	选择是否开启Debug日志,用于排查演练执行过程中遇到的问题。开启Debug后,请到~/logs/chaosblade/chaosblade.log路径下查看日志。

### Java方法调用延迟

模拟Java指定方法调用延迟。

参数名称	是否必选	默认值	参数说明
类名	是	无	完整的类名,包含包名。例如:com.alibaba.service.XxxService。如果模拟接口故障,需填写接口的实现类。
方法名	是	无	方法名,例如:方法getUser(Long userld),则填写getUser。如果存在多个重载方法,如:getUser(String name) 和getUser(Long userld),则对两个方法均生效。
延迟时间	是	无	延迟的时长,单位ms。
延迟波动范围	是	无	延迟时间的上下波动范围,例如:延迟时间为2 ms, 波动范围为1 ms,则实际延迟时间为[1-3] ms。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
进程ID		无	进程的ID。
受影响的请求数	否	0	限制最多发生故障的请求总数,每生效一次故障计数加1,累计发生故障请求数超出设定值后,请求则不再发生故障。填写数值小于等于0时,则表示不限制。
受影响的请求占比			限制发生故障的请求数占所有应该发生故障请求数的百分比,也可代表每次请求发生故障的概率。填写小于或等于0,则表示100%发生故障。
(%)	否	0	⑦ 说明 仅填写百分比数字部分即可,即 80%,填写80。
请求过滤规则	否	无	通过脚本方式自定义规则,通过自定义规则决策是否对请求产生故障。自定义规则生效前提为需满足其它设定条件。
过滤规则执行阶段	否	无	自定义过滤规则执行的阶段,可选择Java方法调用前 执行或Java方法调用后执行。
开启Debug	否	False	选择是否开启Debug日志,用于排查演练执行过程中 遇到的问题。开启Debug后,请到 ~/logs/chaosblade/chaosblade.log路径下查看日 志。

### Java方法抛出异常

模拟Java指定方法调用时抛出指定异常。

参数名称	是否必选	默认值	参数说明
类名	是	无	完整的类名,包含包名。例如:com.alibaba.service.XxxService。如果模拟接口故障,需填写接口的实现类。
方法名	是	无	方法名,例如: 方法getUser(Long userId), 则填写getUser。如果存在多个重载方法,例如: getUser(String name) 和 getUser(Long userId) ,则对两个方法均生效。
异常	是	无	抛出的异常的完成类名,抛出的异常类必须是 java.lang.Exception的子类。例如: java.io.lOException。
	_	chaosblade-	指定抛出异常时的异常信息,默认为chaosblade-mock-exception。
异常信息	否	mock- exception	② 说明 异常信息中请勿包含空格。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
进程ID		无	进程的ID。
受影响的请求数	否	0	限制最多发生故障的请求总数,每生效一次故障计数加1,累计发生故障请求数超出设定值后,请求则不再发生故障。填写数值小于等于0时,则表示不限制。
			限制发生故障的请求数占所有应该发生故障请求数的百分比,也可代表每次请求发生故障的概率。填写小于或等于0,则表示100%发生故障。
受影响的请求占比(%)	否	0	② <b>说明</b> 仅填写百分比数字部分即可,即 80%,填写80。
请求过滤规则	否	无	通过脚本方式自定义规则,通过自定义规则决策是 否对请求产生故障。自定义规则生效前提为需满足 其它设定条件。
过滤规则执行阶段	否	无	自定义过滤规则执行的阶段,可选择Java方法调用前执行或Java方法调用后执行。
开启Debug	否	False	选择是否开启Debug日志,用于排查演练执行过程中遇到的问题。开启Debug后,请 到~/logs/chaosblade/chaosblade.log 看日志。

故障演练·演练场景 应用高可用服务

#### Java方法抛出第一个异常

模拟Java指定方法调用时抛出方法声明中的第一个异常(按照声明顺序)。

#### 参数说明如下:

参数名称	是否必选	默认值	参数说明
类名	是	无	完整的类名,包含包名。例如:com.alibaba.service.XxxService。如果模拟接口故障,需填写接口的实现类。
方法名	是	无	方法名,例如: 方法getUser(Long userId),则填写getUser。如果存在多个重载方法,例如: getUser(String name)和 getUser(Long userId) ,则对两个方法均生效。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找 到唯一进程,使用 ps -ef   grep <key> 来 尝试查找进程,能找到唯一进程则正确。</key>
进程ID		无	进程的ID。
受影响的请求数	否	0	限制最多发生故障的请求总数,每生效一次故障计数加1,累计发生故障请求数超出设定值后,请求则不再发生故障。填写数值小于等于0时,则表示不限制。
受影响的请求占比(%)	否	0	限制发生故障的请求数占所有应该发生故障请求数的百分比,也可代表每次请求发生故障的概率。填写小于或等于0,则表示100%发生故障。  ② 说明 仅填写百分比数字部分即可,即80%,填写80。
请求过滤规则	否	无	通过脚本方式自定义规则,通过自定义规则决策 是否对请求产生故障。自定义规则生效前提为需 满足其它设定条件。
过滤规则执行阶段	否	无	自定义过滤规则执行的阶段,可选择Java方法调用前执行或Java方法调用后执行。
开启Debug	否	false	选择是否开启Debug日志,用于排查演练执行过程中遇到的问题。开启Debug后,请到~/logs/chaosblade/chaosblade.log路径下查看日志。

### 8.3.3. JVM注入动态脚本

向指定的Java方法注入一段动态代码,您可通过代码方式实施任意故障场景,例如篡改方法入参、篡改方法 返回值等。

参数说明如下:

参数名称	是否必选	默认值	参数说明
脚本类型	否	Java	动态脚本的语言类型,可选项:Java、 Groovy。
脚本名称	否	无	动态脚本的名称。
脚本内容	是	无	动态脚本的代码。
类名	是	无	完整的类名,包含报名。例如:com.alibaba.service.XxxService。如果模拟接口故障,需填写接口的实现类。
方法名	是	无	方法名,例如:方法getUser(Long userld),则填写getUser。如果存在多个重载方法,如:getUser(String name)和getUser(Long userld),则对两个方法均生效。
返回值	是	无	指定方法的期望返回值,仅支持基础类型、基础类型的包装类、String类型。如果返回空值,填写null。
进程关键字	必选其一	无	用于识别唯一的关键字,可以通过该关键字查找到唯一进程,使用 ps -ef   grep <key> 来尝试查找进程,能找到唯一进程则正确。</key>
进程ID		无	进程的ID。
受影响的请求数	否	0	限制最多发生故障的请求总数,每生效一次故障计数加1,累计发生故障请求数超出设定值后,请求则不再发生故障。填写数值小于等于0时,则表示不限制。
受影响的请求占比 (%)	否	0	限制发生故障的请求数占所有应该发生故障请求数的百分比,也可代表每次请求发生故障的概率。填写小于或等于0,则表示100%发生故障。 ② 说明 仅填写百分比数字部分即
请求过滤规则	否	无	可,即80%,填写80。 通过脚本方式自定义规则,通过自定义规则决策是否对请求产生故障。自定义规则生效前提
过滤规则执行阶段	否	无	为需满足其它设定条件。 自定义过滤规则执行的阶段,可选择Java方法 调用前执行或Java方法调用后执行。
开启Debug	否	false	选择是否开启Debug日志,用于排查演练执行过程中遇到的问题。开启Debug后,请到 ~/logs/chaosblade/chaosblade.log 路径下查看日志。

故障演练· 演练场景 应用高可用服务

#### 脚本编写说明(以Java为例)

被故障注入代码:

```
public class UserService {
    private UserMapper userMapper;
    public UserDO getUserById(Long userId) {
        UserDO user = userMapper.findUser(userId);
        return user;
    }
}
```

注入的脚本代码:示例脚本演示通过自定义脚本方式篡改方法返回值。

```
// 需要创建类,并import需要的类。
// 如果import自定义类,需要保证该类在目标应用系统中存在。import java.util.Map;
import com.aliaba.model.UserDO;
public class UserServiceInterceptor {
   // 必须包含该方法,且该方法的定义不可改变(返回值、类名、参数均不可改变)。
   // 参数context包含的内容参见《脚本入参说明》。public Object run(Map<String, Object> context)
      //获取getUserById方法的实际入参
      //Map的key是getUserById方法的参数列表的索引位置,value是参数值Map<Integer, Object> argum
ents = context.get("params");
      //获取getUserById方法的一个参数,即userId
      Long userId = (Long)arguments.get(0);
      //构建篡改后的方法返回值UserDO mockUser = new UserDO();
      mockUser.setUserId(userId); //使用真实的userId
      mockUser.setUserName("mock user name"); //构造错误的userName
      //返回篡改后的对象return mockUser;
  }
```

#### 脚本入参说明:

⑦ 说明 列表中的对象可通过conext.get(key)获得。例如: key为object,即通过context.get("object")获得。

key	描述	返回值类型
params	方法的参数列表	java.util.Map <integer,object>(M ap的key为拦截的方法的参数列表中 参数的索引值,value为参数取值)</integer,object>
object	调用方法的应用对象	实际拦截的类的实例
method	当前调用方法的实例	java.lang.reflect.Method
return	实际的返回值(仅"选择生效阶段"填写true时有效)	实际拦截的方法的返回

#### 日志输出:

#### 使用slf4i打印日志。

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
Logger logger = LoggerFactory.getLogger("logger_name");
```

## 8.4. Kubernetes 类场景

### 8.4.1. Node演练场景

Kubernetes集群中Node资源故障场景,包含CPU、网络和进程等基础资源类演练场景。每个Node场景下都包含通用的Node筛选参数,用于查找目标Node。

#### 通用参数说明

参数名称	参数说明
节点名称	节点资源名,选择多个节点资源时资源名之间使用逗号分隔。
节点资源标签	节点资源标签,多个标签之间使用逗号分隔,标签之间是与的关系。
限制节点数量	根据限制条件筛选后,最终故障注入故障的节点数量。默认值为0,表示不限制。

#### 基础资源类演练场景

- CPU类演练场景,请参见CPU类场景
- 网络类演练场景,请参见网络类场景

### 8.4.2. Pod演练场景

Kubernetes集群中Pod资源故障场景,包含删除Pod和Pod网络异常场景。每个Pod场景下都包含通用的Pod筛选参数,用于查找目标Pod。

#### 通用参数

参数名称	参数说明
Pod资源名称	Pod资源名称,多个资源名称之间使用逗号分隔,表示选择多个Pod。
Pod资源标签	Pod资源标签,例如app=frontend,多个标签之间使用逗号分隔,标签之间是与的关系。
命名空间	namespace,用于限定Pod的查找范围。
限制Pod数量	限制根据条件筛选后,最终故障注入的Pod数量(0和默认值表示不限制)。 如果填写的数大于根据条件筛选的数量,则最终只展示条件筛选出的数量。

#### 网络延时场景

- 网络延迟场景,请参见网络延迟。
- 网络丢包场景,请参见网络丢包。

故障演练·演练场景 应用高可用服务

• 篡改域名解析场景,请参见篡改域名解析。

## 8.4.3. Container演练场景

Kubernetes集群中Pod资源下的容器故障场景,包含杀容器以及容器内故障场景。每个容器故障场景下都包含通用的容器筛选参数,用于查找目标容器。

#### 通用参数

参数名称	参数说明
Pod资源名称	Pod资源名称,多个资源名称之间使用逗号分隔,表示选择多个Pod。
Pod资源标签	Pod资源标签,例如app=frontend,多个标签之间使用逗号分隔,标签之间是与的关系。
命名空间	namespace,用于限定Pod的查找范围。
容器ID	容器ID, 多个ID之间使用逗号分隔。
容器名称	容器资源名称,多个资源名称之间使用逗号分隔。
限制Pod数量	限制根据条件筛选后,最终故障注入的Pod数量(0和默认值表示不限制)。如果填写的数大于根据条件筛选的数量,则最终只展示条件筛选出的数量。

#### 网络延时场景

- 网络延迟场景,请参见网络延迟
- 网络丢包场景,请参见<mark>网络丢包</mark>
- 篡改域名解析场景,参见篡改域名解析

应用高可用服务 故障演练·强弱依赖治理

# 9.强弱依赖治理

## 9.1. 强弱依赖治理概述

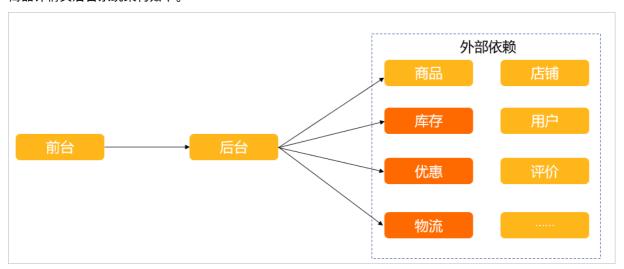
随着分布式微服务的发展,系统正在变得越来越复杂,一个普通的应用也可能依赖了很多其他的服务。在没有明确强弱依赖关系的前提下,系统很难进行限流降级、优化改造等操作。强弱依赖治理就是通过科学的手段持续稳定地得到应用间依赖关系、流量、强弱等数据,提前发现因为依赖问题可能导致的故障,避免依赖故障影响用户体验,积累数据持续推进系统稳定性提升。

#### 什么是强弱依赖?

异常发生时,不影响核心业务流程,不影响系统可用性的依赖称作弱依赖,反之为强依赖。



商品详情页后台系统架构如下。



如果商品详情页对下游依赖是强依赖,例如当下游依赖库存、优惠、物流出现故障的时候,将导致业务流程无法推进,会出现类似如下的说明,严重影响用户体验。



故障演练· 强弱依赖治理 应用高可用服务

● 如果商品详情页对下游依赖是弱依赖,例如当下游依赖**评价、店铺**等系统出现故障的时候,将导致商品详情页部分内容加载不全,出现如下图的情况,但核心业务流程仍可继续推进,用户体验并未受到太多影响。



通过以上对比可以看出,当下游依赖出现问题时,当前系统会受到一些影响,严重影响用户体验的是强依赖,影响较小的则是弱依赖。

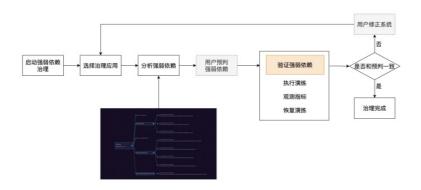
#### 强弱依赖治理

强弱依赖治理就是提前发现因为依赖问题可能导致的故障,避免依赖故障影响用户体验,积累数据持续推进 系统稳定性提升。

强弱依赖模型,包含依赖关系、流量、强弱。依赖治理就是通过科学的手段持续稳定地拿到应用间依赖关系、流量、强弱等数据,将此数据用于系统改造、故障决策等场景。

#### 强弱依赖治理方案

强弱依赖治理每次选中1个应用进行治理,以30天为治理周期。



强弱依赖治理主要包含以下步骤:

- 1. 应用接入(需要安装探针)。
- 2. 依赖分析。
- 3. 依赖预判。
- 4. 依赖验证(通过演练进行验证)。
- 5. 方案归档。

#### 强弱依赖治理的应用

强弱依赖治理主要可以被应用到以下场景:

● **系统改造验收**:对于分布式系统,至少在运行态中,不会因为依赖的系统后台出现故障,引起当前应用出现系统级可用性的故障,例如进程挂掉、频繁FullGC、负载飙高等,何时何地都应具备快速止血的能力。

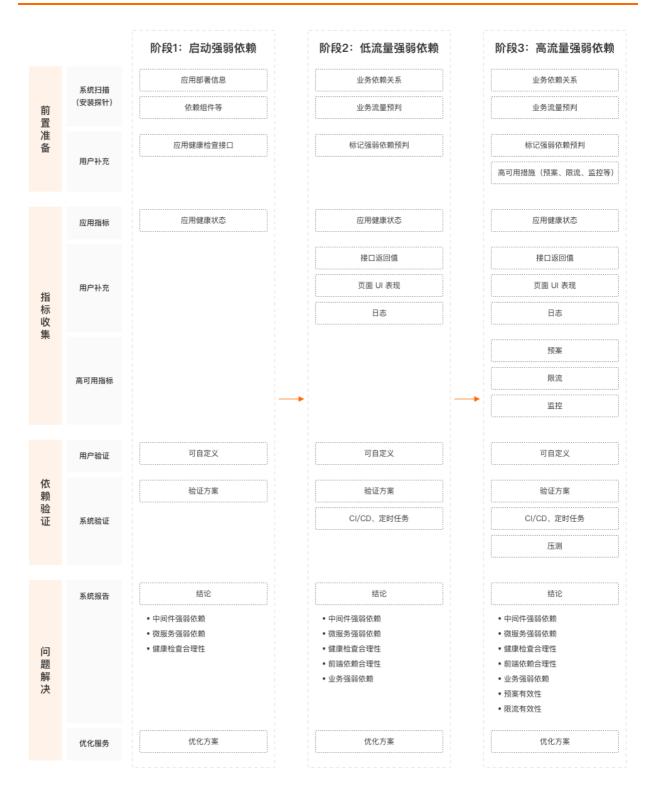
● **限流降级参考**:对于弱依赖,一般都要配置限流或是自动降级策略。比起通过经验值来设定限流值,还是通过实际的依赖验证来进行微调更为可靠。例如对于下游出现的超时情况,就可以通过实验得出基于线程池限流的数据。

- **应用启动顺序**:理想情况下,应用启动更应该做到零强依赖启动,但实际情况是无法做到零强依赖。因此应用启动的依赖顺序也需要实时关注,特别是新IDC、机房建站时,会有着蜘蛛网一样的依赖关系,此时最好是通过系统方式获得依赖顺序。
- **故障根源定位**:后台系统的故障,往往通过上一级的业务故障表现出来。故障处理讲究的是争分夺秒, 良好的强弱依赖,对于系统自动化诊断有非常大的助力作用。
- **依赖容量评估**:正常调用链路下的系统容量需要评估。例如当某个弱依赖挂掉时,需要注意整体的容量是否有变化。

#### 强弱依赖治理的演进

强弱依赖治理分为三阶段: 启动强弱依赖、低流量强弱依赖、高流量强弱依赖。

故障演练·强弱依赖治理 应用高可用服务



#### 参考文档

强弱依赖治理最佳实践

## 9.2. 应用强弱依赖治理

应用强弱依赖治理提供应用层面的强弱依赖关系分析。用户可以选择单个治理应用,查看其依赖分析,并根据依赖验证结果操作演练,完成依赖治理,治理周期为30天。本文介绍应用强弱依赖治理的操作步骤。

#### 前捍冬件

#### ᄞᄣᅏᄄ

需要安装探针才能识别到应用,如未安装请参见管理探针。

#### 背景信息

如果您想快速分析应用与所依赖的各种方法之间的强弱依赖关系,请参见微服务强弱依赖治理。

#### 创建强弱依赖治理

- 1. 登录AHAS控制台, 在左侧导航栏选择故障演练 > 演练方案。
- 2. 在左侧导航栏选择演练方案 > 微服务演练。
- 3. 在微服务演练页面,单击左侧强弱依赖治理。
- 4. 在强弱依赖治理页面,单击创建治理方案,进入应用接入配置向导页。

参数	描述
方案名称	输入治理方案的名称。
治理应用	选择要治理的应用,每个应用在同一个时段只能参与一个治理方案。若未找到应用,可能是由于暂未接入 所需治理的应用,需要单击 <b>新应用接入</b> ,详情请参 见接入演练应用。
机器分组	选择好要治理的应用,系统会自动检测应用所在机器 分组及探针状态。若探针存在异常的情况,可根据提 示查看如何解决。详情请参见 <mark>探针安装常见问题</mark> 。

- 5. 单击下一步,在强弱依赖治理以30天为治理周期对话框单击确定,进入依赖分析配置向导页。
- 6. 在**依赖分析**页面,系统会自动对应用进行依赖分析。在分析过程中,您可以单击**进行分析**区域的**启动**或暂停来控制分析时间。
  - ⑦ 说明 需要保持5分钟正常的业务流量用于系统采集业务数据、分析依赖关系。随着数据的采集会逐步加载分析对象及其依赖对象,可同时选中多个分析对象进行分析。



7. 单击**下一步**,进入**依赖预判**配置向导页,设置强弱依赖关系。 您可以根据业务实际情况,标注分析对象与依赖对象的强弱依赖关系,作为业务预判。

故障演练· 强弱依赖治理 应用高可用服务



8. 单击下一步, 进入依赖验证配置向导页, 单击去验证。



根据上一步的依赖预判,系统自动生成验证用例(演练配置)。

9. 在**去验证前的参数确认**对话框的**分析对象**区域,选择**机器信息**,单击**确定验证**触发验证执行。通过观察实际的监控数据、业务现象,判断真实业务的强弱依赖关系。当依赖预判与验证结果一致时为**符合预期**,否则为**不符合业务预期**。

#### ? 说明

- 随着业务的变化,部分依赖关系会失效。您可以根据业务的实际变化,多次触发依赖分析, 更新依赖预判。
- 每次单击**去验证**时,系统会自动检测**第二步:依赖分析**是否有更新;此外,还会检测验证 用例(演练配置)参数是否有更新。如有变化,系统会自动提示您。
- 。 在验证列表中:
  - 结论存在-或不符合预期,则方案的治理结果为不符合预期。
  - 结论都是符合预期或已失效,则方案的治理结果为符合预期。
- 10. 单击页面下方的**方案归档**,在**您确定要归档此方案吗?** 弹窗中单击**确认归档**,结束1个应用的强弱依赖治理。归档后将无法进行再次分析、验证,但治理报告可正常下载。未手工归档的方案,系统将在30天内自动归档。
  - ? 说明 强弱依赖治理方案以30天为治理周期。

#### 参考文档

强弱依赖治理最佳实践

微服务强弱依赖治理

#### 相关文档

### 9.3. 微服务强弱依赖治理

在应用强弱依赖治理的基础上,微服务强弱依赖治理功能,通过接入MSE,可分析出应用所依赖的Dubbo、Spring Cloud服务,并能够快速的发起方法级别的故障注入,帮助您快速分析出应用与所依赖的各种方法之间的强弱依赖关系。本文介绍如何进行微服务强弱依赖治理。

#### 前提条件

. . . . . . . . . . . .

- 安装故障演练探针,详情请参见管理探针。
- 开通微服务引擎MSE并为应用接入探针,详情请参见应用接入MSE。

#### 操作步骤

- 1. 登录AHAS控制台,在左侧导航栏选择**故障演练 > 演练方案**。
- 2. 在左侧导航栏选择演练方案 > 微服务演练。
- 3. 在微服务演练页面,单击强弱依赖治理。



- 4. 在强弱依赖治理页面,单击创建治理方案。
- 5. 在应用接入配置向导页,输入应用接入信息,然后单击下一步。



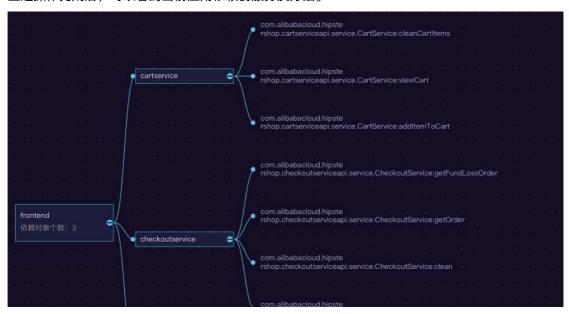
设置项	描述
治理应用	选择要治理的应用,每个应用在同一个时段只能参与一个治理方案。若未找到应 用,可能是由于暂未接入所需治理的应用,需要单击 <b>新应用接入</b> 。详情请参见 <mark>接入 演练应用</mark> 。
机器分组	选择待治理的应用,系统会自动检测应用所在机器分组及探针状态。若探针存在异常的情况,可根据提示查看如何解决。详情请参见探针安装常见问题。

#### 6. 在依赖分析配置向导页:

i. 在**分析维度**区域选中**接口维度**,并按相应指示去接入MSE。具体操作请参见ACK微服务应用接入MSE治理中心。

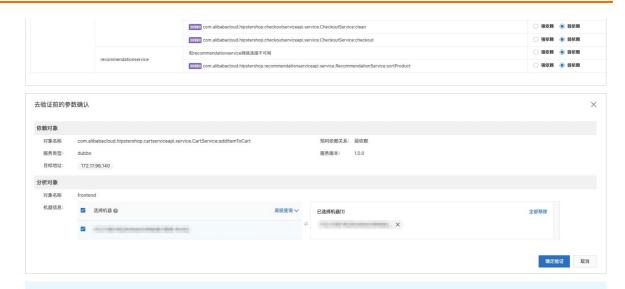


- ii. 接入MSE之后,选择对应的集群。
- iii. 对治理应用所在的namespace开启服务治理。
- iv. 接入完成后,单击**重新分析**。 上述操作完成后,可以看到当前应用依赖的服务及方法。



7. 在依赖预判配置向导页,设置强弱依赖关系,然后单击下一步。





#### ? 说明

- 您可以根据业务实际情况,标注分析对象与依赖对象的强弱依赖关系,作为业务预判。
- 这里的依赖对象既能以整个应用为对象,又能以该应用中的某个服务的方法为对象。
- 8. 在依赖验证配置向导页,单击去验证,然后单击下一步。



#### ? 说明

- 根据上一步的依赖预判,系统自动生成验证用例(演练配置),您可以单击**去验证**触发验证 执行,通过观察实际的监控数据、业务现象,判断真实业务的强弱依赖关系。
- 当依赖预判与验证结果一致时为符合预期,否则为不符合业务预期。
- 随着业务的变化,部分依赖关系会失效。您可以根据业务的实际变化,多次触发依赖分析, 更新依赖预判。
- 每次单击**去验证**时,系统会自动检测第二步的依赖分析是否有更新;其次,检测验证用例 (演练配置)参数是否有更新。如有变化,系统会自动提示您。
- 在验证列表中:
  - 结论存在短划线(-)或不符合预期,则方案的治理结果为不符合预期。
  - 结论都是**符合预期**或**已失效**,则方案的治理结果为符合预期。
- 9. 在方案归档配置向导页,单击方案归档。

故障演练·强弱依赖治理 应用高可用服务

#### ? 说明

- 。 归档后将无法进行再次分析、验证, 但治理报告可正常下载。
- 。 未手工归档的方案, 系统将在30天内自动归档。

# 10.消息演练

### 10.1. 什么是消息演练

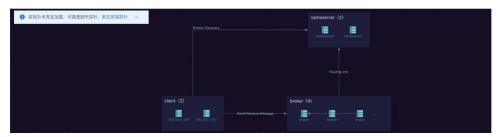
随着应用规模的扩大,系统变得越来越复杂,不可避免地会走向分布式化。各种中间组件会相继被引入系统,其中分布式消息服务更是系统中必不可少的一环。这些消息服务之前并没有在系统中实际经历过真实流量考验,其中某些隐患或缺陷很难被发现,急需通过故障演练来评测高可用性。AHAS提供了强大且高灵活度的故障演练功能,可以根据不同的场景注入对应的故障类型。为了使您的演练服务更便捷,故障演练还提供消息演练功能。本文介绍消息演练的基本概念。

#### 消息演练概述

目前市面上有Rocket MQ、Kafka、Rabbit MQ等流行的消息中间件服务,消息演练功能将这些常见的中间件进行了统一抽离,为消息服务安排分配了精选后的演练最佳实践。您不需要自己去研究分析每种消息服务需要什么演练场景,仅需选择演练方案即可,如下图所示。



在您授权后,消息演练功能会展示对应的消息服务整体架构图,方便您进行演练对象选择,如下图所示。



选择完成后,消息演练功能将会自动生成演练,省去您选择故障和填写参数的步骤,即可进行演练。

#### 支持的部署模式

目前消息演练支持ECS自建服务的部署模式。

#### 消息演练治理方案

您可以每次选择已有的消息组件进行演练,以30天为治理周期。消息演练治理的过程如下图所示。



故障演练·消息演练

#### 消息演练治理控制台的操作主要包含以下步骤:

- 1. 选择演练组件。
- 2. 选择演练方案(需要安装探针)。
- 3. 演练影响预判。
- 4. 演练验证(通过演练进行验证)。
- 5. 方案归档。

#### 消息演练的应用场景

消息演练可以应用于以下场景:

- 评估组件架构: 观测已部署的消息组件当前架构是否符合之前的设计方案,保证正确的组件服务。
- 验证服务依赖:通过演练的故障注入,检测系统对消息组件的依赖程度,探索消息失效的情况下,自身服务是否能够运行正常。
- 降级备案方案:评估消息服务在故障情况下,降级预案是否运行正常,上下游通知服务是否具有容错机制。
- 定位故障根源:经过消息演练之后,积累了消息系统的处理经验,对于未来的系统诊断定位有参考和提升 作用。

#### 消息的演练场景

消息服务通常是由不同的模块组成,例如Rocket MQ,分为生产者Producer、消费者Consumer、服务集群Broker,注册中心NameServer四大模块。单个或多个模块都可能会有不同类型的故障发生,导致模块不能正常运行,例如Producer无法发送消息,这种场景叫做消息的演练场景。需要注意的是演练场景仅是组件表象,其内部故障原因可能是多种多样的,需要进一步进行分析。

# 11.容灾演练

## 11.1. 同城容灾演练产品概述

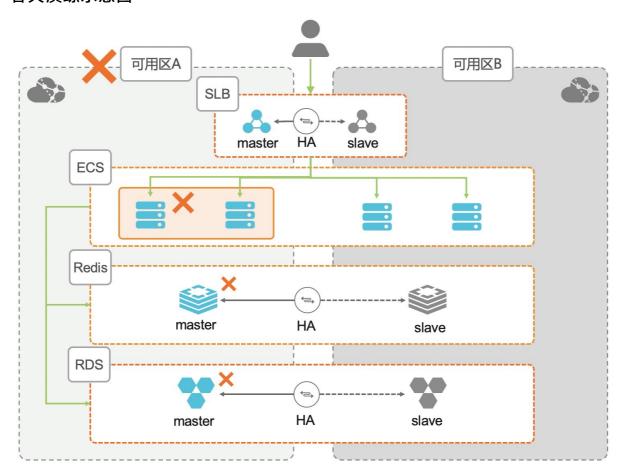
当企业的业务迁移部署在阿里云上,可使用阿里云的云解析DNS(Alibaba Cloud DNS)、负载均衡 SLB(Server Load Balancer)、关系型数据库RDS(Relational Database Service)等产品,搭建同地域多可用区容灾系统架构,实现同城容灾。通过AHAS同城容灾演练验证同城容灾架构设计的合理性与正确性。

#### 适用场景

同城灾备演练适用的灾备场景为:

- 公共云形态:企业业务使用阿里云同地域的不同可用区搭建的同城容灾架构,保障容灾能力。
- 应用级:企业希望对整体的应用做容灾备份演练,而非单独的数据库或存储。
- 云上同城灾备:应对公共云上某地域可用区故障场景。例如:
  - 企业正使用的云产品实例不可用。
  - 可用区的某产品的集群级别的性能衰减或不可用。
  - 基础设置故障导致的整个可用区故障。

#### 容灾演练示意图



#### 同城容灾架构评估项

评估项	要求
SLB	多可用区SLB实例。
应用层	ECS多可用区部署,相关应用多可用区部署。
数据库	以RDS为例,实例为多可用区部署。
缓存	以Redis为例,实例为多可用区部署。
应用设计	应用设计的参考标准如下:      无状态化。     分布式集群设计、避免单点逻辑出现。     幂等。     异步化、松耦合、可中断、可重试。     面向失败的架构设计。

#### 断网演练对业务的影响介绍

演练之前需对自身的部署架构进行充分评估。符合同城容灾架构相关标准只是具有同城容灾能力的基础条件,实际是否可以同城容灾还需通过做容灾演练加以验证。

断网演练执行后,相关指标、业务流量应该会有明显的下跌; "容灾执行"或者"断网恢复"后,相关指标、业务流量应该会有明显的回升。

#### 断网演练的流程

断网演练的一般流程如下:

1. 开启演练:由演练平台执行演练开启。

2. 灾备切换:核心系统或组件进行自动容灾切换。

3. 人工评估: 用户核对业务侧核心指标实际值。

4. 演练恢复: 由演练平台执行演练恢复。

5. 灾备回切:核心系统或组件进行自动灾备回切。

6. 预期评判:结合业务侧核心指标预期值和实际值的差距,判断演练"是否符合预期"。

#### 常见的名词解释

名词	说明
可用区	可用区(Availability Zone,简称AZ)是指在同一地域内,电力和网络互相独立的物理区域。同一可用区内实例之间的网络延时更小。在同一地域内可用区与可用区之间内网互通,可用区之间能做到故障隔离。是否将实例放在同一可用区内,主要取决于对容灾能力和网络延时的要求。
断网范围	实施断网演练的具体范围,涉及可用区内的一个或者多个 云产品、应用。

名词	说明
RPO (Recovery Point Objective)	数据恢复点目标,以时间为单位。即在灾难发生时,系统和数据必须恢复的时间点要求。RPO标志系统能够容忍的最大数据丢失量。系统容忍丢失的数据量越小,RPO的值越小。
RTO (Recovery Time Objective)	恢复时间目标,以时间为单位。即在灾难发生后,信息系统或业务功能从停止到必须恢复的时间要求。RTO标志系统能够容忍的服务停止的最长时间。系统服务的紧迫性要求越高,RTO的值越小。

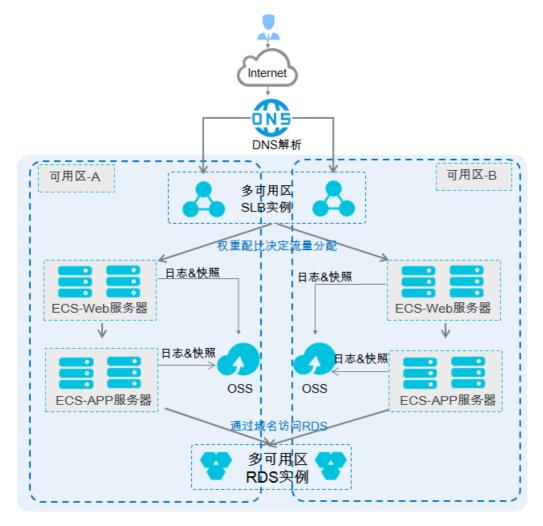
## 11.2. 同城容灾架构概述

本文介绍公共云同城容灾架构的基本原理和接入标准。

#### 基本原理

同城容灾是在相隔较远的两地(同城)建立两套功能相同的IT系统,当一处系统因意外(火灾、地震等)停止工作时,整个应用系统可以切换到另一处,使得该系统功能可以继续正常工作。

对于中大型企业来说,如果希望业务系统具备同城容灾的能力,推荐使用如下图所示的公共云同城容灾架构。



故障演练· 容灾演练 应用高可用服务

#### 架构说明:

● 在同城不同可用区之间对原有应用架构做一套完整的备份,SLB、ECS、RDS、Redis等均在两个机房同时部署。

- 前端部署DNS解析,如果某个可用区出现像IDC机房断电或者火灾等机房级故障时,可以通过前端切换DNS来及时恢复业务。
- 非机房级故障(某个机房的单产品故障,如其中一个机房的ECS服务器损坏),可以通过对单产品的灾备设计来保障发生故障时系统及时切换。

#### 应用设计相关参考标准

应用设计相关参考标准如下:

- 无状态化。
- 分布式集群设计、避免单点逻辑出现。
- 幂等。
- 异步化、松耦合、可中断、可重试。
- 面向失败的架构设计。

### 11.3. 容灾演练断网方式说明

容灾演练提供三种断网方式:可用区断网、安全组断网和AHAS探针方式断网。这三种断网方式爆炸半径依次降低,对断网范围的控制越来越精细。

#### 可用区断网

当整个机房的外部网络出现中断后,外部请求无法进入故障机房,机房内业务同样无法访问外部网络,但机房内网络互通,形成一个局部环境。可用区断网正是为了模拟此类真实故障而提供的机房级故障模拟能力。

可用区断网方式采用租户隔离技术,在用户指定VPC网络下对可用区交换机层面进行网络访问控制(网络ACL),实现子网流量的受控访问。

- 机房级业务整体不可用。
- 机房级云服务不可用。
- 机房级云基础设施不可用。
- 可用区断网基于VPC网络ACL的高级特性。若VPC不支持高级特性,则无法使用可用区断网方式。更多信息,请参见VPC高级功能。
- 如果您可用区内的ECS实例绑定了辅助弹性网卡,且辅助弹性网卡绑定了设置网卡可见模式的EIP,那么可用区断网不过滤该ECS实例的流量。
- 可用区断网不会影响SLB或RDS等高可用实例的心跳检查,无法自动触发相关高可用实例的主备切换功能。

#### 使用场景

#### 安全组断网

安全组断网是一种借助阿里云专有网络安全组技术实现的轻量级且精细化的断网方式,可以针对用户的 ECS、Redis及RDS实例进行断网模拟。

针对ECS实例的断网模拟采用的是变更安全组策略的方式。如果ECS绑定了多块辅助网卡,每一块辅助网卡对应的安全组策略同样会进行故障注入,故障实验恢复时,系统会还原ECS绑定的原始安全组策略及辅助网卡对应的原始安全组。可用区断网在交换机级别进行模拟,而安全组断网在实例级别进行模拟。

针对Redis及RDS实例的断网模拟是通过主动触发主备切换的方式实现的。在实例进行主备切换时,业务可能会出现长连接闪断的情况,以此验证业务是否具备失效重连机制。

- 部分业务或大规模业务实例不可用。
- Redis、RDS云服务高可用实例部分不可用。
- 安全组断网只针对新建立的网络连接进行阻断,并不会阻断已建立的连接,因此可能会存在断网不彻底的问题。
- 部分地域存在极少量机器安全组断网不生效的问题。

#### AHAS探针方式断网

AHAS探针断网方式是应用高可用服务中故障演练平台自带的探针断网方式。用户需要在ECS中安装探针,然后通过探针进行本地网络受控访问。探针断网方式支持Host以及Kubernetes模式,相比于安全组断网,可实现更加彻底的断网效果,在故障模拟生效后,任何请求均可能会被拒绝(白名单机制中保留了访问部分云服务及SSH通道,防止意外无法恢复问题)。其次,相比于新建网络延迟故障,AHAS探针方式断网在参数自动填充、流程自动编排以及多网卡自适应等层面均实现了自动化处理,简化产品配置使用。

- 部分业务出现不可用,且需要模拟彻底断网。
- Redis、RDS云服务高可用实例部分不可用。

故障演练·容器演练

# 12.容器演练

## 12.1. 什么是容器报警演练

容器演练是针对Kubernetes集群定制的故障演练,可以对Kubernetes集群中至关重要的高可用特性进行验证。目前AHAS仅支持容器演练中的报警演练。本文主要介绍容器演练中的报警演练功能。

#### 报警演练功能概述

Kubernet es集群的自动报警功能是保障Kubernet es集群高可用的重要功能之一。当集群的API Server监测到特定事件,报警模块会通过邮件、短信等方式通知用户处理。

AHAS容器演练中的监控报警验证功能通过模拟事件触发,验证集群监控报警设置是否正常可用,并统计报警成功率、以及事件触发到监控报警的平均延迟,从而验证Kubernetes集群监控报警系统的有效性。

#### 报警演练功能特性

- 无损。不同于其他故障演练模块对系统注入真实故障,报警演练只会对集群的API Server投递事件,由事件触发报警链路,不会对系统造成真实故障。
- 无侵入。无需在用户集群中安装探针以注入故障,所有事件投递都是通过API Server来进行网络请求。

### 12.2. 报警演练

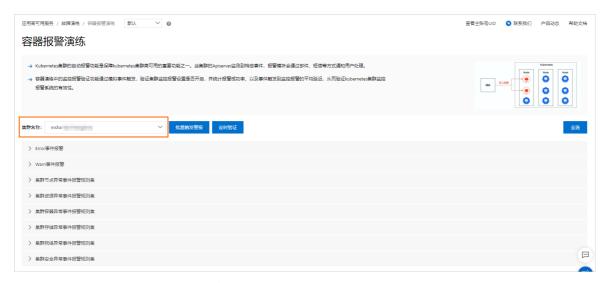
容器演练中的报警演练功能通过触发模拟事件,验证集群监控报警设置是否开启,并统计报警成功率、以及事件触发到监控报警的平均延迟,从而验证Kubernet es集群监控报警系统的有效性。AHAS的报警演练支持批量触发报警和定时触发报警,本文介绍这两种报警演练的操作方法。

#### 前提条件

集群已开启报警配置,具体操作,请参见容器服务报警管理。

#### 批量触发报警

- 1. 登录AHAS控制台,然后在页面左上角选择地域。
  - ② 说明 目前容器演练暂不支持公网地域。
- 2. 在控制台左侧导航栏选择故障演练 > 演练方案。
- 3. 在左侧导航栏选择演练方案 > 容器演练。
- 4. 在容器报警演练页面,从集群名称下拉列表中选择演练的集群。



5. 在容器报警演练页面,选择待验证的报警事件(可多选),然后单击批量触发警报。

报警演练开始后AHAS会立即投递报警事件,验证通知用户的报警链路是否正常,计算报警延迟时间。 您可以通过**状态**列查看报警是否成功触发。



□ 注意 具体的报警联系人和历史信息,您可以在<mark>容器服务管理控制台</mark>中查看和设置。具体操作,请参见<mark>容器服务报警管理</mark>。

#### 定时触发报警

定时报警验证可以在不触发真实报警的情况下验证报警系统的有效性。

- 1. 登录AHAS控制台,然后在页面左上角选择地域。
  - ② 说明 目前容器演练暂不支持公网地域。
- 2. 在控制台左侧导航栏选择故障演练 > 演练方案。
- 3.
- 4. 在容器报警演练页面,选择待验证的报警事件(可多选),然后单击定时验证。
- 5. 在规则配置对话框,配置定时报警规则,然后单击确认。

参数	说明
启动定时报警	<ul><li>选择是,开启定时报警。</li><li>选择否,关闭定时报警。</li></ul>
每日触发时间	在下拉列表中选择每日触发报警的具体时间,精确到分。

参数	说明
钉钉机器人webhook地址	输入自定义的钉钉机器人Webhook地址,用于通知报警事件。关于Webhook地址的具体操作,请参见如何获取自定义机器人Webhook地址?。
定时验证规则列表	显示已配置定时报警演练的规则。

# 13.演练空间

### 13.1. 演练空间概述

演练空间用于管理成员和演练,管理内容包括演练方案、演练执行和成员等。演练空间便于您根据业务,组织演练和管理团队。本文介绍演练空间中的基本概念。

#### 演练空间分类

演练空间分为我管理的空间和我参与的空间,根据权限的不同您可以在不同的空间做相应的演练和用户管理操作。我管理的空间中有一种特殊的空间叫**我的空间**。

#### 我的空间

每个用户均有一个**我的空间**,该用户即为此空间的管理员。**我的空间**是一个特殊空间的存在,空间名称不能被修改,只能叫做**我的空间**,且任何其他成员都不能被邀请进入**我的空间。我的空间**包含您所有创建的演练,您可以对空间中的演练进行创建、删除、编辑、执行等操作。

#### 我管理的空间

我管理的空间是您自定义创建的空间。作为管理员,您可以对此空间进行以下操作:

- 新增、删除空间,编辑空间的基本信息。
- 新增、移除、拷贝演练。
- 新增、删除成员,并设置各个成员的权限。

#### 我参与的空间

我参与的空间是您作为成员参与的空间。作为成员,您可以对此空间进行以下操作:

- 查看演练。
- 编辑演练(只有管理员设置该账号有此权限才可以编辑)。
- 执行演练(只有管理员设置该账号有此权限才可以执行)。

#### 空间权限说明

您在各种空间的角色和权限说明如下。

空间分类	角色	编辑空间名	管理成员	演练操作
我的空间	管理员	不支持	不支持	支持
我管理的空间	管理员	支持	支持	支持
我参与的空间	成员	不支持	不支持	根据管理员赋予成 员的权限不同而支 持的操作不同。

## 13.2. 管理演练空间

演练空间可以统计您执行过的演练数据,便于您根据不同的业务,组织不同演练,管理各个演练和用户。本文介绍如何管理演练空间。

#### 背景信息

关于演练空间概念的更多信息,请参见演练空间概述。

故障演练·演练空间 应用高可用服务

#### 管理我的空间

**我的空间**是一个特殊的空间,每个账号都有一个**我的空间**且空间名称不能被编辑,您也不能邀请其他成员 进入**我的空间**。

- 1. 登录AHAS控制台,然后在页面左上角选择地域。
- 2. 在左侧导航栏选择故障演练 > 我的空间。

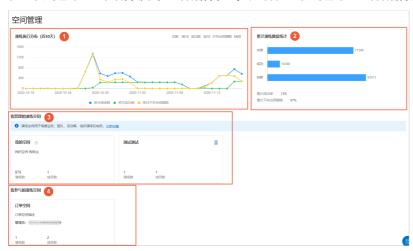


您可以在我的空间页面进行以下操作:

- ①:空间基础信息区域,可以编辑空间描述。
- ②:执行概要区域,可以查看到**我的空间**中演练执行的统计数据,包括此账号执行过的演练数、失败的演练数等。
- ③:演练列表区域,可以新建演练、执行演练、拷贝演练、删除演练等。新建演练的具体操作,请参见创建演练。

#### 查看演练空间

- 1. 登录AHAS控制台,然后在页面左上角选择地域。
- 2. 在左侧导航栏选择故障演练 > 数据管理 / 然后在左侧导航栏选择数据管理 > 空间管理。



您可以在空间管理页面进行以下操作:

- o ①:演练执行分布区域,可以查看近30天内此账号演练执行的总数、成功数以及不符合预期数。
- ②:**累计演练数据统计**区域,可以查看此账号所有执行过的演练的总数、失败的个数、成功的个数以及成功率。
- ③:**我管理的演练空间**区域,可以查看此账号具有管理员角色的演练空间,作为空间管理员,您可

以创建新的空间,也可以删除已有空间(除我的空间外)。

○ ④: 我参与的演练空间区域,可以查看此账号具有成员角色的演练空间,成员可以根据已有的权限 对演练进行查看、拷贝或移除的操作。

#### ? 说明

- RAM用户演练执行的分布图和累计演练数据统计图与阿里云账号一致,展示的是阿里云账号的统计信息。
- 不同环境(Namespace)展示的演练空间相同。环境管理的更多信息,请参见<mark>管理环境</mark>。

#### 管理成员

在**我管理的演练空间**(除**我的空间**外)内您可以管理成员,可以在我管理的空间新增、删除成员,以及编辑成员的权限。

- 1. 登录AHAS控制台,然后在页面左上角选择地域。
- 2. 在左侧导航栏选择故障演练 > 空间管理。
- 3. 单击我管理的一个演练空间(除我的空间外)。
- 4. 在演练空间详情页的**基本信息**区域,单击**管理成员**,在**管理成员**对话框,可以新增、删除成员或编辑成员的权限。

权限	可执行的操作
查看	可执行以下操作: o 查看空间。 o 查看执行概要。 o 查看演练。
编辑	可执行以下操作: o 查看空间。 o 查看执行概要。 o 查看、编辑、创建、克隆、添加、移除演练。  ⑦ 说明 编辑权限包含了查看权限。
执行	可执行以下操作: o 查看空间。 o 查看执行概要。 o 查看、执行演练。  ② 说明 执行权限包含了查看权限。

#### 相关信息

关于演练的具体操作,请参见以下文档:

- 创建演练
- 执行演练
- 停止演练