阿里云
专有云大数据版

安全白皮书

产品版本：V2.1.0

文档版本：20180730

为了无法计算的价值 | 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
	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。	**禁止**： 重置操作将丢失用户配置数据。
	该类警示信息可能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，恢复业务所需时间约10分钟。
	用于警示信息、补充说明等，是用户必须了解的内容。	**注意**： 导出的数据中包含敏感信息，请妥善保存。
	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。	**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。	单击 **确定**。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *`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。	`swich` *`{stand | slave}`*

目录

法律声明.....I
通用约定.....I
1 安全白皮书介绍.....1
2 安全责任共担.....2
2.1 阿里云安全责任.....2
2.2 用户安全责任.....2
3 阿里云专有云安全架构.....4
3.1 云平台安全架构.....4
3.1.1 基础设施安全.....4
3.1.1.1 物理安全.....4
3.1.1.2 服务器设备安全.....5
3.1.1.3 网络设备安全.....5
3.1.1.4 基础网络安全.....5
3.1.2 云操作系统安全.....6
3.1.2.1 虚拟化安全.....6
3.1.2.2 基础系统服务安全.....7
3.1.2.3 系统管理和调度安全.....8
3.1.3 云存储安全.....8
3.1.3.1 访问控制.....8
3.1.4 大数据计算安全.....8
3.1.4.1 授权管理.....8
3.1.4.2 跨项目空间的资源分享.....9
3.1.4.3 数据保护机制.....9
3.1.5 数据安全.....9
3.1.5.1 数据安全体系.....9
3.1.5.2 数据所有权.....9
3.1.5.3 多副本冗余存储.....10
3.1.5.4 镜像管理.....10
3.1.5.5 残留数据清除.....10
3.1.5.6 运维数据安全.....10
3.1.6 云产品代码安全.....10
3.2 云用户（租户）侧安全.....11
3.2.1 账户安全.....11
3.2.2 主机安全.....12
3.2.3 应用安全.....12
4 专有云云产品安全.....13

4.1 运维权限管理（OAM）.....13
4.2 天基权限管理（数据中心管理）.....13
4.3 访问控制RAM.....13
4.4 对象存储OSS.....14
4.4.1 安全隔离.....14
4.4.2 鉴权认证.....14
4.4.2.1 身份验证.....14
4.4.2.2 权限控制.....15
4.4.2.3 RAM和STS支持.....15
4.4.3 数据安全.....16
4.4.4 传输加密.....16
4.4.4.1 服务器端加密.....16
4.4.4.2 客户端加密.....16
4.4.4.3 KMS加密.....16
4.4.5 日志审计.....17
4.4.6 防盗链.....17
4.5 MaxCompute.....17
4.5.1 安全隔离.....17
4.5.2 权鉴认证.....18
4.5.2.1 身份验证.....18
4.5.2.2 权限控制.....18
4.5.2.3 RAM支持.....25
4.5.3 数据安全.....25
4.5.4 传输加密.....26
4.5.5 日志审计.....26
4.5.6 访问控制-IP白名单.....26
4.5.7 MaxCompute支持VPC.....34
4.6 分析型数据库AnalyticDB.....34
4.6.1 安全隔离.....34
4.6.2 鉴权认证.....34
4.6.2.1 身份验证.....35
4.6.2.2 权限控制.....35
4.6.2.3 RAM和STS支持.....36
4.6.3 数据安全.....36
4.6.4 日志审计.....37
4.6.5 VPC支持.....37
4.7 关系网络分析I+.....37
4.7.1 安全隔离.....37
4.7.2 鉴权认证.....38
4.7.2.1 身份验证.....38
4.7.2.2 权限控制.....38

4.7.3 数据安全.....38
4.7.4 传输加密.....38
4.7.5 日志审计.....38
4.7.6 系统安全.....38
4.7.6.1 漏洞扫描机制.....38
4.7.6.2 安全漏洞更新修复方案.....39
4.7.6.3 系统防御机制.....39
4.7.7 基础设施安全.....39
4.7.8 等保认证.....39
4.8 流计算StreamCompute.....39
4.8.1 账号安全.....39
4.8.2 业务安全.....39
4.8.3 数据安全.....40

1 安全白皮书介绍

数据安全和用户隐私是阿里云专有云最重要的原则，阿里云致力于打造公共、开放、安全的专有云云计算服务平台。通过技术创新，不断提升计算能力与规模效益，将云计算变成真正意义上的基础设施。

阿里云专有云竭诚为用户提供稳定、可靠、安全、合规的云计算基础服务，帮助保护用户的系统及数据的可用性、机密性和完整性。

本白皮书介绍了阿里云专有云云安全体系，主要包括下列内容：

- 安全责任共担
- 安全合规和隐私
- 专有云平台架构安全
- 专有云各云产品提供的安全功能
- 专有云云盾提供的安全服务

同时，本白皮书提供了安全使用阿里云产品和云盾安全产品的最佳实践来帮助您更好地使用阿里云专有云平台以及理解安全控制整体环境。

2 安全责任共担

基于专有云平台的用户应用，其安全责任由双方共同承担：阿里云确保专有云平台本身架构的安全性，用户负责专有云平台运营以及基于专有云平台构建的应用系统的安全。

阿里云

阿里云负责专有云飞天分布式云操作系统及之上的各种云服务产品本身的安全，从而为用户提供高可用和高安全的专有云平台。同时，专有云基于阿里巴巴集团多年攻防技术积累，为用户提供云盾安全服务，进一步保障用户的专有云环境的安全。

用户

用户负责以安全的方式配置和使用专有云平台以及MaxCompute等云产品，并基于这些云产品以安全可控的方式构建自己的应用。同时，用户可使用专有云云盾安全产品及服务为其专有云环境提供安全防护。

2.1 阿里云安全责任

阿里云负责分布式云操作系统及云服务产品本身的安全，并为用户提供保护专有云平台、云端应用及数据的技术手段。

- 保障专有云云平台架构安全
- 提供及时发现专有云云平台的安全漏洞并修复（修复漏洞过程不影响业务可用性）的安全服务及技术
- 提供协助用户与外部第三方独立安全监管与审计机构合作，对阿里云专有云进行安全合规与审计评估的服务
- 为用户提供保护云端信息系统的技术手段
- 为用户提供安全审计手段
- 为用户提供数据加密手段
- 为用户提供云盾安全服务

2.2 用户安全责任

用户基于阿里云提供的专有云平台构建自己的云端应用系统，综合运用专有云产品的安全功能、云盾安全产品及服务保护自己的专有云环境。

用户应妥善管理专有云环境中的账户，为每个运维管理人员授予完成运维管理工作需要的最小权限，通过群组授权实现职责分离。同时，通过操作审计服务记录管理控制台操作及OpenAPI调用日志。

对于专有云提供大数据计算服务（MaxCompute），用户需要管理这些服务的账户及授权，并使用这些服务提供的安全功能。例如，配置RDS服务的源IP白名单。

3 阿里云专有云安全架构

阿里云为专有云设计了多个层面的纵深防御安全体系，包括基础设施安全、云操作系统安全、云存储安全、大数据计算安全、数据安全、云产品代码安全、安全审计、云平台安全运营服务等云平台层面的安全架构保障；以及账户安全、主机安全、安全运营服务等云用户（租户）层面的安全架构保障。

3.1 云平台安全架构

3.1.1 基础设施安全

3.1.1.1 物理安全

对于专有云机房物理安全方面的要求，主要包括但不限于双路供电、访问控制、视频监控、火灾检测、热备机房等安全措施。

双路供电

为保障业务7*24小时持续运行，专有云的数据中心机房的每一个负载均由两个电源供电，两个电源之间可以进行切换。若电源发生故障，在其中一个电源失电的情况下可以投切到另一个电源供电。

访问控制

对于专有云数据中心的物理设备和机房的访问要具备访问控制，包括机房的进出访问控制。例如，对于进出机房或者携带设备进出机房，物理设备的配置、启动、关机、故障恢复等，均需具备相应的访问控制策略。

视频监控

专有云数据中心机房应装设视频监控系统或者有专人24小时值守，对通道等重要部位进行监视。例如，对出入通道进行视频监控，同时报警设备应该能与视频监控系统或者出入口控制设备联动，实现对于监控点的有效监视。

火灾检测

专有云数据中心机房应配备火灾自动报警系统，包括火灾自动探测器、区域报警器、集中报警器和控制器等。火灾自动报警系统能够对于火灾发生的部位以声、光或点的形式发出报警信号，并启动自动灭火设备，切断电源、关闭空调设备等。

热备机房

在故障发生时，按照预先设定的故障恢复方案，使用热备份单元自动替换故障单元，实现故障的自动恢复。

3.1.1.2 服务器设备安全

阿里云对专有云物理服务器本身的安全进行加固，主要包括但不限于账号安全、文件权限、系统服务、主机入侵检测系统等方面。

账号安全

针对物理服务器账号的口令长度、复杂度、密码长度、口令生命期进行安全策略设置，删除空口令的账号，设置登录超时（TIMEOUT）时间等。

文件权限

针对重要目录进行完整性监控，在黑客篡改和写入文件时，能第一时间发现入侵行为。

系统服务

禁用物理服务器上不必要的系统服务，减少服务器的受攻击面。

3.1.1.3 网络设备安全

账号安全

针对网络设备的账号口令策略、密码配置文件的存储加密进行安全加固。

- 为网络设备建立只读账号，只允许查看配置，实现读、改配置的账号分离。
- 通过集中管控策略，实现账号的统一管理。
- 采用多因素认证的方式保障网络设备的账号安全。

服务

禁用网络设备上的服务，减少网络设备的受攻击面；并且禁用与网络设备不相关的功能。

日志集中化

将网络设备产生的日志进行集中化收集和管理。

3.1.1.4 基础网络安全

微隔离

专有云平台对专有云网络环境中的管理网络（OPS）、业务网络、物理网络进行了三网安全隔离。OPS网络、业务网络、物理网络三张网络之间通过网络访问控制策略实现三网逻辑隔离，彼此

之间不能互相访问。同时，采取网络控制措施防止非授权设备私自连接云平台内部网络，并防止云平台物理服务器主动外连。

IP、MAC、ARP防欺骗

在传统网络环境中，IP、MAC、ARP欺骗一直是网络面临的严峻考验。通过IP、MAC、ARP欺骗，黑客可以扰乱网络环境，窃听网络机密。专有云平台通过物理服务器上的网络底层技术机制，彻底解决地址欺骗问题。

专有云平台在物理服务器数据链路层隔离由服务器向外发起的异常协议访问，阻断服务器的MAC、ARP欺骗，并在宿主机网络层防止服务器IP欺骗。

网络入侵检测

专有云平台部署网络入侵检测系统，实时发现网络中的异常行为并告警。网络入侵检测系统的功能包括HTTP异常入侵检测、HTTP漏洞发现等。同时，对于部分系统服务进行安全漏洞检测，包括但不限于Redis、MongoDB、MySQL等服务。

3.1.2 云操作系统安全

3.1.2.1 虚拟化安全

虚拟化技术是云计算平台的主要技术支撑，通过计算虚拟化、存储虚拟化、网络虚拟化来保障云计算环境下的多租户隔离。阿里云的虚拟化安全技术主要包括租户隔离、补丁热修复、逃逸检测三大基础安全部分来保障专有云平台虚拟化层的安全。

租户隔离

虚拟化管理层在租户隔离中起到至关重要的作用。基于硬件虚拟化技术的虚拟机管理，将多个计算节点的虚拟机在系统层面进行隔离，租户不能访问相互之间未授权的系统资源，从而保障计算节点的基本计算隔离。同时，虚拟化管理层还提供存储隔离和网络隔离。

- **计算隔离**

  专有云平台提供各种基于云的计算服务，包括各种计算实例和服务，同时支持自动伸缩以满足应用程序及各用户的需求。这些计算实例和服务从多个级别提供计算隔离以保护数据，同时保障用户需求的配置灵活性。计算隔离中关键的隔离边界是管理系统与用户虚拟机之间、以及用户虚拟机之间的隔离，这种隔离由Hypervisor直接提供。在专有云平台使用的虚拟化环境中，将用户实例作为独立的虚拟机运行，并且通过使用物理处理器权限级别强制执行隔离，确保用户虚拟机无法通过未授权的方式访问物理主机和其他用户虚拟机的系统资源。

- **存储隔离**

作为云计算虚拟化基础设计的一部分，阿里云将基于虚拟机的计算与存储分离。这种分离使得计算和存储可以独立扩展，从而更容易提供多租户服务。在虚拟化层，Hypervisor采用分离设备驱动模型实现I/O虚拟化。虚拟机所有I/O操作都会被Hypervisor截获并处理，保证虚拟机只能访问分配给它的物理磁盘空间，从而实现不同虚拟机硬盘空间的安全隔离。用户实例服务器释放后，原有的磁盘空间将会被可靠地清零以保障用户数据安全。

- **网络隔离**

  虚拟网络是建立在物理网络结构之上的逻辑结构，每个逻辑虚拟网络与所有其他虚拟网络隔离。

逃逸检测

虚拟机逃逸攻击主要包括两个基本步骤：首先将攻击方控制的虚拟机置于与其中一个攻击目标虚拟机相同的物理主机上；然后破坏隔离边界，以窃取攻击目标的敏感信息或实施影响攻击目标功能的破坏行为。

专有云虚拟化管理程序通过使用高级虚拟机布局算法以防止恶意用户的虚拟机运行在特定物理机上。同时，阿里云在虚拟化管理软件层面还提供了虚拟化管理程序加固、虚拟化管理程序下攻击检测、虚拟化管理程序热修复三大核心技术来防范恶意虚拟机的攻击。

补丁热修复

专有云虚拟化平台支持补丁热修复技术，通过补丁热修复技术使得系统缺陷或者漏洞的修复过程不需要用户重启系统，从而不影响用户业务。

3.1.2.2 基础系统服务安全

飞天操作系统

- **分布式文件系统安全**

  分布式文件系统使用三副本技术，将系统中的数据保存三份。如果其中一份副本丢失，系统会自动进行三副本的拷贝操作，始终保持拥有三份副本。同时，根据安全策略，三份副本不会存储在同一个物理存储介质上，保持存储的分离操作。

  所有访问分布式文件系统的操作，必须通过Capability认证，只有携带了允许的Capability才能与系统进行通信，从而解决未经授权访问的操作。

  存储在分布式文件系统中的数据，采用二进制格式化存储的方式，避免直接查看到明文信息，造成信息泄露。

- **远程过程调用模块安全**

远程过程调用模块在飞天操作系统进行通信时，采用指定的二进制格式进行通信，保证传输过程中的效率以及传输的安全，保证即使数据被中间人劫持也无法还原数据。

- **任务调度模块安全**

  任务调度模块采用沙箱的方式对程序进行隔离。

基础设施

针对NTP、DNS服务部署DDoS攻击防护、DNS区域传送、DNS放大攻击防御、NTP放大攻击防御等安全措施，保障NTP和DNS服务器的安全。

3.1.2.3 系统管理和调度安全

专有云平台管理系统采用Docker容器化的部署方式。由阿里云安全专家对云平台管理系统进行SDL安全审核，通过代码审核、线上测试、需求分析、威胁建模的方式，保障云平台管理系统的整体安全性。

3.1.3 云存储安全

3.1.3.1 访问控制

对云存储服务的资源访问分为拥有者访问和第三方用户访问：拥有者是指存储空间（bucket）的拥有者，第三方用户是指访问该bucket资源的其他用户。访问方式分为匿名访问和带签名访问：如果请求中没有携带任何与身份相关的信息即为匿名访问；带签名访问是指按照云存储服务API规定在请求头部或者在请求URL中携带签名的相关信息的请求。

3.1.4 大数据计算安全

3.1.4.1 授权管理

项目空间（Project）是专有云平台大数据计算服务实现多租户体系的基础，是用户管理数据和计算的基本单位。当用户申请创建一个项目空间之后，该用户就是这个空间的所有者（Owner）。也就是说，这个项目空间内的所有对象（如表、实例、资源、UDF等）都属于该用户。除了Owner之外，任何人都无权访问此项目空间内的对象，除非获得Owner的授权许可。

当项目空间的Owner决定对另一个用户授权时，Owner需要先将该用户添加到自己的项目空间中，只有添加到项目空间中的用户才能够被授权。

角色（Role）是一组访问权限的集合。当需要对一组用户赋予相同的权限时，可以使用角色来授权。基于角色的授权可以大大简化授权流程，降低授权管理成本。当需要对用户授权时，应当优先考虑是否应该使用角色来完成。

大数据计算服务支持对项目空间里的用户或角色，针对Project、Table、Function、Resource Instance四种对象，授予不同权限。

3.1.4.2 跨项目空间的资源分享

假设用户是项目空间的Owner或管理员（admin角色），其它用户需要申请访问用户的项目空间资源。如果申请人属于该用户的项目团队，建议用户使用项目空间的用户与授权管理功能；如果申请人并不属于该用户的项目团队，可以使用基于Package的跨项目空间的资源分享功能。

Package是一种跨项目空间共享数据及资源的机制，主要用于解决跨项目空间的用户授权问题。使用Package后，A项目空间管理员可以对B项目空间需要使用的对象进行打包授权（也就是创建一个Package），然后许可B项目空间安装这个Package。在B项目空间管理员安装Package后，就可以自行管理Package是否需要进一步授权给自己Project下的用户。

3.1.4.3 数据保护机制

如果项目空间中的数据非常敏感，绝对不允许流出到其他项目空间时，可以使用项目空间保护机制（设置ProjectProtection）。明确要求该项目空间中的数据只能流入，不能流出。

3.1.5 数据安全

3.1.5.1 数据安全体系

阿里云数据安全体系从数据安全生命周期角度出发，采取管理和技术两方面的手段，进行全面、系统的建设。通过对数据生命周期（数据生产、数据存储、数据使用、数据传输、数据传播、数据销毁）各环节进行数据安全管理管控，实现数据安全目标。

专有云平台在数据安全生命周期的每一个阶段，都有相应的安全管理制度以及安全技术保障。

3.1.5.2 数据所有权

2015年7月，阿里云发起中国云计算服务商首个“数据保护倡议”，这份公开倡议书明确：运行在云计算平台上的开发者、公司、政府、社会机构的数据，所有权绝对属于用户；云计算平台不得将这些数据移作它用。平台方有责任和义务，帮助用户保障其数据的私密性、完整性和可用性。

3.1.5.3 多副本冗余存储

专有云使用分布式存储技术，将文件分割成许多数据片段分散存储在不同的设备上，并且将每个数据片段存储多个副本。分布式存储不但提高了数据的可靠性，也提高了数据的安全性。

3.1.5.4 镜像管理

专有云平台的云服务器提供快照与自定义镜像功能，快照可以保留某个时间点上的系统数据状态，用于数据备份，便于用户快速实现灾难恢复。用户可以使用快照创建自定义镜像，将快照的操作系统、数据环境信息完整地包含在镜像中。快照采用增量方式，两个快照之间只有数据变化的部分才会被拷贝。

3.1.5.5 残留数据清除

对于曾经存储过用户数据的内存和磁盘，一旦释放和回收，其上的残留信息将被自动进行零值覆盖。

3.1.5.6 运维数据安全

运维人员未经用户许可，不得以任意方式访问用户未经公开的数据内容。

专有云平台遵循生产数据不出生产集群的原则，从技术上控制了生产数据流出生产集群的通道，防止运维人员从生产系统拷贝数据。

3.1.6 云产品代码安全

在云产品安全生命周期（SPLC）中，阿里云安全专家在各个开发节点中都进行严格审核并评估代码的安全性，保障阿里云提供给用户的产品的代码安全。

云产品安全生命周期（Secure Product Lifecycle，简称SPLC）是阿里云为云上产品量身定制的云产品安全生命周期，目标是将安全融入到整个产品开发生命周期中。SPLC在产品架构审核、开发、测试审核、应急响应的各个环节层层把关，每个节点都有完整的安全审核机制确保产品的安全性能够满足严苛的云上要求，从而有效地提高云产品的安全能力并降低安全风险。整个云产品安全生命周期可以分为六大阶段：产品立项、安全架构审核、安全开发、安全测试审核、应用发布、应急响应。

- **在产品立项阶段**，安全架构师和产品方一同根据业务内容、业务流程、技术框架建立功能需求文档（FRD）、绘制详细架构图，并在阿里云产品上云的所有安全基线要求中确认属于产品范围的《安全基线要求》。同时，本阶段会安排针对性的安全培训课程与考试给产品方人员，从而避免在后续产品开发中出现明显的安全风险。

- **在安全架构审核阶段**，安全架构师在上一阶段产出的FRD和架构图的基础上对产品进行针对性的安全架构评估并做出产品的威胁建模。在威胁建模的过程中，安全架构师会对产品中的每一个需要保护的资产、资产的安全需求、可能的被攻击场景做出详细的模型，并提出相对应的安全解决方案。安全架构师会综合《安全基线要求》和威胁建模中的安全解决方案，一并与产品方确认对于该产品的所有《安全要求》。
- **在安全开发阶段**，产品方会根据《安全要求》在产品开发中遵守安全编码规范，并实现产品的相关安全功能和要求。为了保证云产品快速持续的开发、发布与部署效率，产品方会在本阶段进行自评确认《安全要求》都已经实现，并提供相对应的测试信息（如代码实现地址，自测结果报告等）给负责测试的安全工程师，为下一阶段的安全测试审核做好准备。
- **在安全测试审核阶段**，安全工程师会根据产品的《安全要求》对其进行架构设计、服务器环境等全方位的安全复核，并对产品的代码进行代码审核和渗透测试。在此阶段发现的安全问题会要求产品方进行安全修复和加固。
- **在应用发布阶段**，只有经过安全复核并且得到安全审批许可后，产品才能通过标准发布系统部署到生产环境，以防止产品携带安全漏洞在生产环境运行。
- **在应急响应阶段**，安全应急团队会不断监控云平台可能的安全问题，并通过外部渠道（如ASRC等）或者内部渠道（如内部扫描器、安全自测等）得知安全漏洞。在发现漏洞后应急团队会对安全漏洞进行快速评级，确定安全漏洞的紧急度和修复排期，从而合理分配资源，做到快速并合理的修复安全漏洞，保障阿里云用户、自身的安全。

3.2 云用户（租户）侧安全

专有云在用户侧安全提供了多个层面的安全保障，其中包括了账户安全、主机安全、云盾、以及安全运营服务。

3.2.1 账户安全

专有云平台提供多种安全机制来帮助用户保护账户安全，防止未授权的用户操作。这些安全机制包括云账户登录、创建子用户、集中管理子用户权限、数据传输加密、子用户操作审计等，用户可以使用这些机制来保护云账户的安全。

3.2.2 主机安全

入侵检测

专有云平台用户可以通过在主机上配置云盾安骑士客户端，实现与云盾安全中心的联动防护，获取入侵检测的安全能力。主机的入侵检测包括异地登录提醒、识别暴力破解攻击、网站后门查杀、主机异常检测等功能。

漏洞管理

专有云平台用户可以通过在主机上配置云盾安骑士客户端，实现与云盾安全中心的联动防护，获取漏洞管理的安全能力。主机的漏洞管理综合了多套扫描引擎（网络端、本地端、PoC验证），全面批量检测出系统存在的所有漏洞，并提供一键修复、生成修复命令、一键批量验证功能，实现漏洞管理的闭环。

3.2.3 应用安全

Web应用防护

通过Web应用防火墙，防御SQL注入、XSS跨站脚本、常见Web服务器插件漏洞、木马上传、非授权核心资源访问等OWASP常见攻击，过滤海量恶意访问，避免网站资产数据泄露，保障网站应用的安全与可用性。

代码安全

在云产品安全生命周期（SPLC）中，阿里云的安全专家在各个开发节点中都会严格审核和评估代码的安全性，从而保障阿里云提供给用户的产品的代码安全质量。同时，阿里云强烈建议企业用户对其上线的应用进行黑白盒代码安全检测，务求上线后的应用不会存在安全漏洞，增加用户本身的业务的安全强壮性。

4 专有云云产品安全

4.1 运维权限管理（OAM）

运维权限管理系统（Operation Administrator Manager，简称OAM）是Apsara Stack运维系统的权限管理平台。OAM采用一种简化的基于角色的访问控制（RBAC）模型，管理员可以通过OAM为运维人员授予角色，运维人员依据各自的角色，对各运维系统拥有相应的操作权限。

4.2 天基权限管理（数据中心管理）

天基是一套自动化的数据中心管理系统，管理专有云数据中心的硬件生命周期与各类静态资源，包括程序、配置、操作系统镜像、数据等。

天基为飞天系统及专有云各种产品的应用及服务提供了一套通用的版本管理、部署以及热升级方案，使得基于天基的服务在大规模分布式的环境下达到自动化运维的效果，极大地提高运维效率，并提高系统可用性。

权限管理

天基的权限管理也采用OAM系统。天基用户权限包括天基Admin权限、Project权限和Service权限：

- **Admin权限**：Admin用户可以对整个天基平台的页面进行操作。
- **Project权限**：
  - 普通用户需要由管理员开通Project权限，才能查看天基平台中**运维** > **Project**中的Project信息。
  - 普通用户需要由管理员开通Project权限，才能查看天基平台中**运维** > **集群管理**中的集群信息并执行该节点下的相关操作。
- **Service权限**：普通用户需要由管理员开通Service权限，才能查看天基平台中**运维** > **Service** > **运维**中的服务信息并执行该节点下的相关操作。

4.3 访问控制RAM

云租户可以使用RAM建立主子账号体系。

访问控制管理（Resource Access Management，简称RAM）是专有云平台为用户提供的用户身份管理与访问控制服务。通过RAM，可以创建、管理用户账号（比如员工、系统或应用程序），并可

以分配这些用户账号对其名下资源具有的操作权限。当存在多用户协同操作资源时，使用RAM可以避免与其他用户共享云账号密码或访问密钥，按需为用户分配最小权限，从而降低信息安全风险。

4.4 对象存储OSS

对象存储服务（Object Storage Service，简称OSS）提供海量、安全、低成本、高可靠的云存储服务。OSS可以被理解成一个即开即用，无限大空间的存储集群。相比传统自建服务器存储，OSS在可靠性、安全性、成本和数据处理能力方面都有着突出的优势。使用OSS，用户可以通过网络随时存储和调用包括文本、图片、音频和视频等在内的各种非结构化数据文件。

OSS将数据文件以对象/文件（object）的形式上传到存储空间（bucket）中。用户可以进行以下操作：

- 创建一个或者多个存储空间
- 每个存储空间中添加一个或多个文件
- 通过获取已上传文件的地址进行文件的分享和下载
- 通过修改存储空间或文件的属性或元信息来设置相应的访问权限
- 通过云控制台执行基本和高级OSS任务
- 通过开发工具包SDK或直接在应用程序中进行RESTful API调用执行基本和高级OSS任务

4.4.1 安全隔离

OSS将用户数据切片，按照一定规则，离散存储在分布式文件系统中，并且用户数据和数据索引分离存储。OSS的用户认证采用Access Key对称密钥认证技术，对于用户的每个HTTP请求都验证签名。在用户验证通过后，重组用户离散存储的数据，从而实现多租户间的数据存储隔离。

4.4.2 鉴权认证

4.4.2.1 身份验证

用户可以在Apsara Stack控制台中自行创建Access Key。Access Key由AccessKey ID和AccessKey Secret组成，其中ID是公开的，用于标识用户身份，Secret是秘密的，用于用户身份的鉴别。

当用户向OSS发送请求时，需要首先将发送的请求按照OSS指定的格式生成签名字符串，然后使用AccessKey Secret对签名字符串进行加密（基于HMAC算法）产生验证码。验证码带时间戳，以防止重放攻击。OSS收到请求以后，通过AccessKey ID找到对应的AccessKey Secret，以同样的方

法提取签名字符串和验证码，如果计算出来的验证码和提供的一致即认为该请求是有效的；否则，OSS将拒绝处理这次请求，并返回HTTP 403错误。

4.4.2.2 权限控制

对OSS的资源访问分为拥有者访问和第三方用户访问。拥有者是指bucket的拥有者，第三方用户是指访问bucket资源的其他用户。访问分为匿名访问和带签名访问。对于OSS来说，如果请求中没有携带任何和身份相关的信息即为匿名访问。带签名访问是指按照OSS API文档中规定的在请求头部或者在请求URL中携带签名的相关信息。

OSS提供bucket和object的权限访问控制。

Bucket有三种访问权限：public-read-write，public-read 和 private。

- public-read-write：任何人（包括匿名访问）都可以对该bucket中的object进行PUT、Get和Delete操作。
- public-read：只有该bucket的创建者可以对该bucket内的object进行写操作（包括Put和Delete Object）；任何人（包括匿名访问）可以对该bucket中的object进行读操作（Get Object）。
- private：只有该bucket的创建者可以对该bucket内的object进行读写操作（包括Put、Delete和Get Object）；其他人无法访问该bucket内的object。

说明：
用户新创建一个bucket时，如果不指定bucket权限，OSS会自动为该bucket设置private权限。

Object有四种访问权限：public-read-write，public-read，private和default。

- public-read-write：所有用户拥有此object的读写权限。
- public-read：非此object的Owner拥有此object的读权限，只有此object的Owner拥有此object的读写权限。
- private：此object的Owner拥有该object的读写权限，其他的用户对此object没有读、写权限。
- default：object遵循bucket的访问权限。

说明：
用户上传object时，如果不指定object权限，OSS会为object设置default权限。

4.4.2.3 RAM和STS支持

OSS支持RAM/STS鉴权。

RAM (Resource Access Management) 是阿里云提供的资源访问控制服务。通过RAM，主账号可以创建出子账号，子账号从属于主账号，所有资源都属于主账号，主账号可以将所属资源的访问权限授予给子账号。

STS（Security Token Service）是阿里云提供的临时访问凭证服务，提供短期访问权限管理。STS可以生成一个短期访问凭证给用户使用，凭证的访问权限及有效期限由用户定义，访问凭证过期后会自动失效。

4.4.3 数据安全

数据在客户端和服务器之间传输时有可能会出错。OSS支持对各种方式上传的Object返回其CRC64值，客户端可以和本地计算的CRC64值作对比，从而完成数据完整性的验证。

OSS对新上传的Object进行CRC64的计算，并将结果存储为Object的元信息，随后在返回的response header中增加**`x-oss-hash-crc64ecma`**头部，表示其CRC64值，该64位CRC根据 *ECMA-182*标准 计算得出。

4.4.4 传输加密

4.4.4.1 服务器端加密

OSS支持在服务器端对用户上传的数据进行加密（Server-Side Encryption）。当用户上传数据时，OSS对收到的用户数据使用AES256进行加密，然后再将加密得到的数据永久保存下来。用户下载数据时，OSS自动对保存的加密数据解密后把原始数据返回给用户，并在返回的HTTP请求Header中声明该数据进行了服务器端加密。

用户创建Object时，只需要在Put Object的请求中携带x-oss-server-side-encryption的HTTP header，并指定其值为AES256，即可以实现该Object的服务器端加密存储。

4.4.4.2 客户端加密

客户端加密（Server-Side Encryption）是指用户数据在发送给远端服务器之前就完成加密，而加密所用的密钥明文只保留在用户本地，从而可以保证用户数据安全，即使数据泄漏别人也无法解密得到原始数据。OSS通过SDK中的函数针对OSS Bucket中的数据进行客户端加密，在本地加密后再上传到OSS Bucket中。

4.4.4.3 KMS加密

阿里云Key Management Service （KMS）是一项将安全、高度可用的硬件和软件相结合，提供可扩展到云端的密钥管理系统的服务。KMS使用客户主密钥（CMK）加密OSS Bucket对象，通过

KMS API集中创建加密密钥，定义策略以控制密钥的使用方法，以及审核密钥使用情况来证明它们使用得当。用户可以利用这些密钥来保护在OSS Bucket中的数据。

4.4.5 日志审计

OSS提供自动保存访问日志记录（logging）功能，用户开启Bucket的日志保存功能后，OSS自动将访问这个Bucket的请求日志，以小时为单位，按照固定的命名规则，生成一个Object写入用户指定的目标Bucket（Target Bucket），作为审计或者特定行为分析使用。请求日志中包含请求时间、来源IP、请求对象、返回码、处理时长等内容。

4.4.6 防盗链

为了防止用户在OSS上的数据被其他人盗链，OSS支持基于HTTP header中表头字段referer的防盗链方法。用户可以通过Apsara Stack控制台或者API的方式对一个Bucket设置referer字段的白名单和是否允许referer字段为空的请求访问。例如，对于一个名为oss-example的Bucket，设置其referer白名单为`http://www.aliyun.com/`。则所有referer为`http://www.aliyun.com/`的请求才能访问oss-example这个Bucket中的Object。

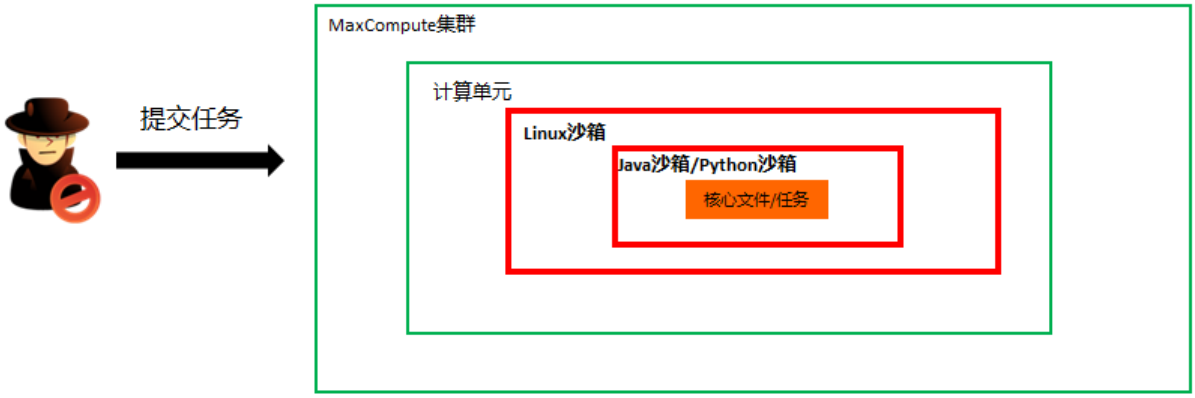
4.5 MaxCompute

4.5.1 安全隔离

MaxCompute支持多租户的使用场景，通过阿里云账号认证体系，即认证方式采用AccessKey对称密钥认证技术，同时对于用户的每一个HTTP请求都会进行签名认证，针对不同的用户数据进行数据存储隔离，用户数据被离散存储在分布式文件系统中。可以同时满足多用户协同、数据共享、数据保密和安全的需要，做到真正的多租户资源隔离。

同时，MaxCompute中所有计算是在受限的沙箱中运行的，多层次的应用沙箱，从KVM级到Kernel级。系统沙箱配合鉴权管理机制，用来保证数据的安全，以避免出现内部人员恶意或粗心造成服务器故障。

图 4-1: 沙箱保护



4.5.2 权鉴认证

4.5.2.1 身份验证

用户可以在云控制台中自行创建AccessKey。AccessKey由AccessKey ID和AccessKey Secret组成，其中AccessKey ID是公开的，用于标识用户身份，AccessKey Secret是秘密的，用于用户身份的鉴别。

当用户向MaxCompute发送请求时，首先需要将发送的请求按照MaxCompute指定的格式生成签名字符串，然后使用AccessKey Secret对签名字符串进行加密以生成请求签名。MaxCompute收到用户请求后，通过AccessKey ID找到对应的AccessKey Secret，以同样的方法提取签名字符串和验证码，如果计算出来的验证码和提供的一致即认为该请求是有效的；否则，MaxCompute将拒绝处理这次请求，并返回HTTP 403错误。

4.5.2.2 权限控制

用户对MaxCompute资源访问分为两种，即用户主账号访问和用户子账号访问。主账号是阿里云的一个账号主体，主账号下可以包含不同的子账号以便用户可以灵活使用。MaxCompute支持主子账号的权限访问策略。

- 当用户使用主账号访问时，MaxCompute会校验该主账号是否为对应资源的所有者，只有对应资源的所有者才具备访问该资源的权限。
- 当用户使用子账号访问时，此时会触发子账号授权策略。MaxCompute会校验该子账号是否被对应主账号授予了访问该资源的权限，同时也会校验该子账号对应的主账号是否具有该资源的所有者权限。

MaxCompute目前支持两种授权机制来完成对子账号的访问权限控制。

- **ACL授权**：ACL授权是一种基于对象的授权。通过ACL授权的权限数据（即访问控制列表，Access Control List）被看做是该对象的一种子资源，只有当对象已经存在时，才能进行ACL授权操作。当对象被删除时，通过ACL授权的权限数据会被自动删除。ACL授权支持的授权方法是采用类似SQL92定义的GRANT/REVOKE命令进行授权，通过对应的授权命令来完成对已存在的项目空间对象的授权或撤销授权。
- **Policy授权**：Policy授权是一种基于策略的授权。通过Policy授权的权限数据（即访问策略）被看做是授权主体的一种子资源。只有当主体（用户或角色）存在时，才能进行Policy授权操作。当主体被删除时，通过Policy授权的权限数据会被自动删除。Policy授权使用MaxCompute自定义的一种访问策略语言来进行授权，允许或禁止主体对项目空间对象的访问权限。

两套体系的关系如下表所示：

表 4-1: ACL授权和Policy授权的关系

ACL授权	Policy授权
每次权限管理操作均是对效果（授权、撤销）、对象（如表、资源等）、主体（用户或是角色）、操作（读、写、删除等）的组合描述，例如“允许用户zinan.tang读取表table1中的数据”。	
N/A	支持条件授权，目前支持20种条件操作。
N/A	支持Allow（允许访问）和Deny（拒绝访问）授权。
授权或撤销授权时，对象和主体必须存在。	支持对“不存在”或“不确定”的对象和主体授权，授权时不会验证授权存在性，支持用通配符描述对象或主体。
删除一个对象时，会自动撤销与该对象关联的授权。	删除一个对象时，系统不会自动修改与该对象关联的Policy。
支持经典的Grant/Revoke语句。	通过上传Policy文本描述授权操作。

MaxCompute还支持更多的访问权限控制机制。

跨项目空间的资源分享

假设用户是项目空间的Owner或管理员（admin角色），有人需要申请访问用户的项目空间资源。如果申请人属于用户的项目团队，此时建议用户使用项目空间的用户授权管理功能。但是如果申请人并不属于用户的项目团队， 此时用户可以使用基于Package的跨项目空间的资源分享功能。

Package是一种跨项目空间共享数据及资源的机制，主要用于解决跨项目空间的用户授权问题。

使用Package之后，A项目空间管理员可以对B项目空间需要使用的对象进行打包授权（也就是创建一个Package），然后许可B项目空间安装这个Package。在B项目空间管理员安装Package之后，就可以自行管理Package是否需要进一步授权给自己Project下的用户。

Package使用方法示例如下。

- Package创建者的操作示例如下。

```
create package <pkgname>
-- 创建Package
```

说明：

- 仅project的owner有权限进行该操作。
- 目前创建的package名称不能超过128个字符。

```
add project_object to package package_name [with privileges
privileges]
remove project_object from package package_name
project_object ::= table table_name |
instance inst_name |
function func_name |
resource res_name
privileges ::= action_item1, action_item2, ...
-- 添加资源到Package
```

说明：

- 目前支持的对象类型不包括Project类型，即不允许通过Package在其他Project中创建对象。
- 添加到Package中的不仅仅是对象本身，还包括相应的操作权限。当没有通过[with privileges privileges]来指定操作权限时，默认为只读权限， 即Read/Describe/Select。 “对象及其权限”被看作一个整体，添加后不可被更新。若有需要，只能删除和重新添加。

```
allow project <prjname> to install package <pkgname> [using label <
number>]
-- 赋予其它项目空间使用权限
```

```
disallow project <prjname> to install package <pkgname>
-- 撤销其它项目空间使用权限
```

```
delete package <pkgname>
-- 删除Package
```

```
show packages
```

```
-- 查看Package列表
```

```
describe package <pkgname>
-- 查看Package详细信息
```

- Package使用者的操作示例如下。

```
install package <pkgname>
-- 安装Package
```

说明：

- 仅project的owner有权限进行该操作。
- 对于安装Package来说，要求**pkgName**的格式为：*<projectName>.<packageName>*。

```
uninstall package <pkgname>
-- 卸载Package
```

说明：

对于卸载Package来说，要求**pkgName**的格式为：*<projectName>.<packageName>*。

```
show packages
-- 查看已创建和已安装的package列表
```

```
describe package <pkgname>
-- 查看package详细信息
```

被安装的Package是独立的MaxCompute对象类型，若要访问Package里的资源（即其他项目空间分享给用户的资源），必须拥有对该Package的Read权限。若请求者无Read权限，则需向ProjectOwner或Admin申请，ProjectOwner或Admin可以通过ACL授权或Policy授权机制来完成授权。

示例如下，**仅供参考**：通过ACL授权允许云账号odps_test@aliyun.com访问Package里的资源。

```
use prj2;
install package prj1.testpkg;
grant read on package prj1.testpackage to user
aliyun$odps_test@aliyun.com;
```

通过Policy授权允许项目空间prj2中任何用户都可以访问Package里的资源。

```
use prj2;
install package prj1.testpkg;
```

```
put policy /tmp/policy.txt;
```

说明：

/tmp/policy.txt的内容如下。

```
{
"Version": "1",
"Statement":
[{
"Effect":"Allow",
"Principal":"*",
"Action":"odps:Read",
"Resource":"acs:odps:*:projects/prj2/packages/prj1.testpkg"
}]
}
```

列级别访问控制

基于标签的安全（LabelSecurity）是项目空间级别的一种强制访问控制策略（Mandatory Access Control, MAC），它的引入可以让项目空间管理员更加灵活地控制用户对列级别敏感数据的访问。

LabelSecurity需要将数据和访问数据的人进行安全等级划分。一般来讲，会将数据的敏感度标记分为如下四类：

- 0级（不保密，Unclassified）。
- 1级（秘密，Confidential）。
- 2级（机密，Sensitive）。
- 3级（高度机密，Highly Sensitive）。

MaxCompute也遵循这一分类方法，ProjectOwner需要定义明确的数据敏感等级和访问许可等级划分标准。默认时所有用户的访问许可等级为0级，数据安全级别默认为0级。

LabelSecurity对敏感数据的粒度可以支持列级别，管理员可以对表的任何列设置敏感度标记（Label），一张表可以由不同敏感等级的数据列构成。而对于view，也支持和表相同的设置，即管理员可以对view设置label等级。View的等级和它对应的基表的label等级是独立的，在view创建时，默认的等级也是0级。

在对数据和人分别设置安全等级标记之后，LabelSecurity的默认安全策略如下：

- No-ReadUp：不允许用户读取敏感等级高于用户等级的数据，除非显式授权。
- Trusted-User：允许用户写任意等级的数据，新建数据默认为0级（不保密）。

说明：

- 在一些传统的强制访问控制系统中，为了防止数据在项目空间内部的任意分发，一般还支持更多复杂的安全策略，例如：不允许用户写敏感等级不高于用户等级的数据（No-WriteDown）。但在MaxCompute平台中，考虑到项目空间管理员对数据敏感等级的管理成本，默认安全策略并不支持No-WriteDown，如果项目空间管理员有类似需求 ， 可以通过修改项目空间安全配置（ SetObjectCreatorHasGrantPermission=false）以达到控制目的。
- 如果是为了控制数据在不同项目空间之间的流动，则可以将项目空间设置为受保护状态（ProjectProtection）。设置之后，只允许用户在项目空间内访问数据，这样可以有效防止数据流出道项目空间之外。

项目空间中的LabelSecurity安全机制默认是关闭的，ProjectOwner需要自行开启。需要注意，LabelSecurity安全机制一旦开启，上述的默认安全策略将被强制执行。此时，当用户访问数据表时，除了必须拥有Select权限外，还必须获得读取敏感数据的相应许可等级。

数据保护机制（Project Protection）

同时在多个项目空间中拥有访问权限的用户，可以自由地使用任意支持跨Project的数据访问操作来转移项目空间的数据。但是，如果项目空间中的数据非常敏感，绝对不允许流出到其他项目空间中去，此时管理员可以使用项目空间保护机制——设置ProjectProtection，明确要求项目空间中**“数据只能本地循环，允许写入，不能读出”**。

具体设置如下：

```
set projectProtection=true
-- 设置ProjectProtection规则为：数据只能流入，不能流出。
```

说明：

需要注意，默认ProjectProtection不会被设置，默认值为false，即数据保护机制按需开启。

开启数据保护机制后的数据流出方法

在用户的项目空间被设置了ProjectProtection之后，用户可能会遇到如下的需求：某人向用户提出申请，因正常的业务需求，需要将某张表的数据导出用户的项目空间。而且经过用户的审查之后，那张表也的确没有泄漏用户关心的敏感数据。此时，为了不影响正常的业务需求，MaxCompute为用户提供了在ProjectProtection被设置之后的两种数据导出途径。

设置ExceptionPolicy

ProjectOwner在设置ProjectProtection时可以附带一个exception策略，命令如下：

```
SET ProjectProtection=true WITH EXCEPTION <policyFile>
```

说明：

此时的policy不同于Policy授权（尽管它与Policy授权语法完全一样），它只是对项目空间保护机制的例外情况的一种描述，即所有符合policy中所描述的访问情形都可以打破ProjectProtection规则。

Exception policy相关示例如下：

```
{
    "Version": "1",
    "Statement":
    [{
        "Effect":"Allow",
        "Principal":"ALIYUN$Alice@aliyun.com",
        "Action":["odps:Select"],
        "Resource":"acs:odps:*:projects/alipay/tables/table_test",
        "Condition":{
            "StringEquals": {
                "odps:TaskType":["DT", "SQL"]
            }
        }
    }]
    }
-- 允许云账号Alice@aliyun.com可以通过SQL任务对表alipay.table_test执行Select
操作时将数据流出到alipay项目空间之外。
```

说明：

- Exception policy并不是一种普通的授权。如果上述示例中，云账号Alice并没有对表alipay.table_test的Select操作权限，那么即使设置了上述exception policy，Alice仍然是无法导出数据。
- ProjectProtection是一种数据流向的控制，而不是访问控制。只有在用户能访问数据的前提下，控制数据流向才是有意义的。

设置TrustedProject

若当前项目空间处于受保护状态，如果将数据流出的目标空间设置为当前空间的TrustedProject，那么向目标项目空间的数据流向将不会被视为触犯ProjectProtection规则。如果多个项目空间之间两两互相设置为TrustedProject，那么这些项目空间就形成了一个TrustedProject Group，数据可以在这个Project Group内流动，但禁止流出到Project Group之外。

管理TrustedProject的命令如下：

```
list trustedprojects;
-- 查看当前project中的所有TrustedProjects。
add trustedproject <projectname>;
-- 在当前project中添加一个TrustedProject。
remove trustedproject <projectname>;
-- 在当前project中移除一个TrustedProject。
```

资源分享与数据保护的关系

在MaxCompute中，基于package的资源分享机制与ProjectProtection数据保护机制是正交的，但在功能上却是相互制约的。

MaxCompute规定：**资源分享优先于数据保护**。即如果一个数据对象是通过资源分享方式授予其他项目空间用户访问，那么该数据对象将不受ProjectProtection规则的限制。

防止数据从项目空间流出的更多检查

如果要防止数据从项目空间的流出，在设置ProjectProtection=true之后，还需检查如下配置：

- 确保没有添加trustedproject。如果有设置，则需要评估可能的风险。
- 确保没有设置exception policy。如果有设置，则需要评估可能的风险，尤其要考虑TOC2TOU数据泄露风险。
- 确保没有使用package数据分享。如果有设置，则需要确保package中没有敏感数据。

4.5.2.3 RAM支持

MaxCompute支持RAM鉴权。

RAM（**Resource Access Management**）是阿里云提供的资源访问控制服务。通过RAM，主账号可以创建出子账号，子账号从属于主账号，所有资源都属于主账号，主账号可以将所属资源的访问权限授予给子账号。

4.5.3 数据安全

专有云提供一个扁平的线性存储空间，并在内部对线性地址进行切片，一个分片称为一个Chunk。对于每一个Chunk，都会复制出三个副本，并将这些副本按照一定的策略存放在集群中的不同节点上，保证用户数据的可靠。

在专有云数据存储系统中，有三类角色，分别称为Master、Chunk Server和Client。MaxCompute用户的每一个写操作经过层层转换，最终会交由Client来执行，执行过程如下：

1. Client计算出这个写操作对应的Chunk。
2. Client向Master查询该Chunk的三份副本的存放位置。
3. Client根据Master返回的结果，向对应的三个Chunk Server发出写请求。
4. 如果三份副本都写成功，Client向用户返回成功；反之，Client向用户返回失败。

Master的分布策略会综合考虑集群中所有Chunk Server的磁盘使用情况、在不同交换机机架下的分布情况、电源供电情况、及机器负载情况，尽量保证一个Chunk的三个副本分布在不同机架下的不同 Chunk Server 上，从而有效防止由于一个Chunk Server或一个机架的故障导致的数据不可用。

当有数据节点损坏，或者某个数据节点上的部分硬盘发生故障时，集群中部分Chunk的有效副本数会小于三。一旦发生这种情况，Master就会启动复制机制，在Chunk Server之间复制数据，保证集群中所有Chunk的有效副本数达到三份。

综上所述，对MaxCompute上的数据而言，所有用户层面的操作都会同步到底层三份副本上，无论是新增、修改还是删除数据。通过这种机制，保障用户数据的可靠性和一致性。

另外，在用户进行删除操作后，释放的存储空间由飞天分布式文件系统回收，禁止任何用户访问，并在被再次使用前进行内容擦除，最大限度保证用户的数据安全性。

4.5.4 传输加密

MaxCompute提供Restful的传输接口，其传输安全性由HTTPS保证。

4.5.5 日志审计

MaxCompute会针对不同用户不同日志数据进行日志审计。在MaxCompute内部，MaxCompute提供元数据仓库进行日志数据存储，包括静态数据、运行记录及安全信息等内容。

- 静态数据：是指一旦产生就不会自动消失的数据。
- 运行记录：表示一个任务的运行过程，该记录只会出现在一个分区中。
- 安全信息：都来自TableStore，用于保存白名单、ACL列表等。

元数据仓库：就是使用MaxCompute来分析MaxCompute自己的运行状况，将MaxCompute中的各种元信息整理汇总成MaxCompute中的表，方便用户查询和统计。

4.5.6 访问控制-IP白名单

MaxCompute安全上的访问控制有多个层次：如项目空间的多租户及安全认证机制，只有获取了正确的经过授权的**`AccessKey ID`**及**`AccessKey Secret`**才能通过鉴权，在已经赋予的权限范围内

进行数据访问和计算。本文主要介绍在以上访问认证基础上增强的一种以IP白名单的方式，进行访问控制的配置方法和策略，并指导用户完成相关配置。

说明：

- 获取需要配置的IP地址的方式如下：
  1. 如果使用MaxCompute Console（odpscmd）在集群内部使用（如ag上使用），可以直接获取机器的IP地址。
  2. 如果使用应用系统（如base或者datax）进行项目空间数据访问，需要配置base或者datax所在的部署server机器的IP地址。
  3. 如果使用了代理服务器或者经过了多跳代理服务器来访问MaxCompute服务实例，需要添加的IP地址为最后一跳代理服务器的IP地址。
  4. 如果是ECS机器中访问MaxCompute服务，获取到的IP地址为NATIP。
- IP地址配置的格式如下：

  多个IP由**“逗号”**分割，且支持三种IP格式：1、单独IP地址。2、IP地址段，由**“-”**连接。3、带有子网掩码的IP。

  示例如下：

```
10.32.180.8,10.32.180.9,10.32.180.10
-- 单独IP地址。
10.32.180.8-10.32.180.12
-- IP地址段。
10.32.180.0/23
-- 带子网掩码的IP地址。
```

下面将分别介绍project group级别IP白名单，project级别IP白名单以及系统级别IP白名单所涉及的相关配置操作。

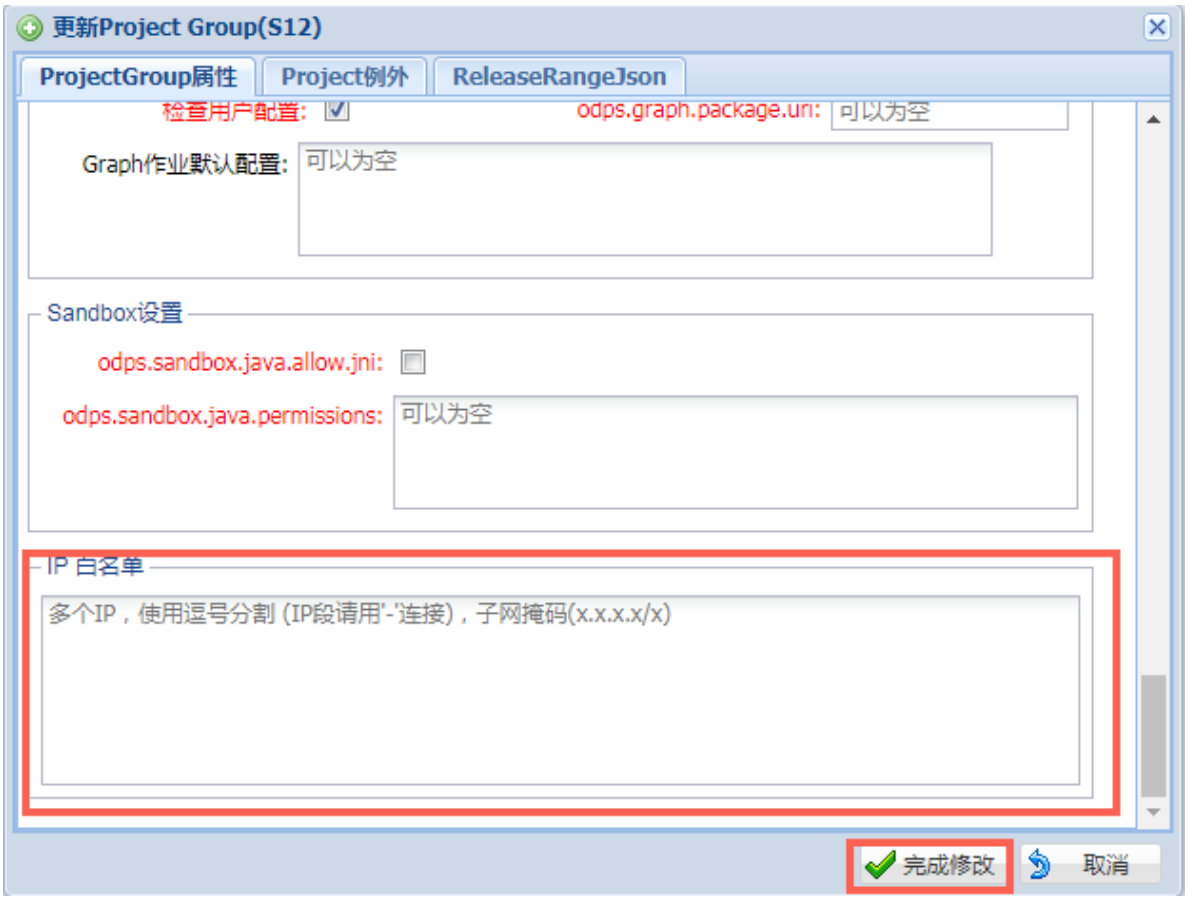
Project group级别IP白名单配置

Project group级别进行白名单控制时， 如果某一个project属于project group，那么在project group中配置白名单后，该project也受此配置限制。

具体的配置方式如下所示。

1. 在**AdminConsole**中选择**ODPS配置** > **Group管理**，选中需要配置的group，双击打开配置框。
2. 在弹出的配置框中完成相关配置后，单击**完成修改**。

图 4-2: Project group级别IP白名单配置1



3. 配置完成后可以在project group属性配置中查看配置结果。

图 4-3: Project group级别IP白名单配置2

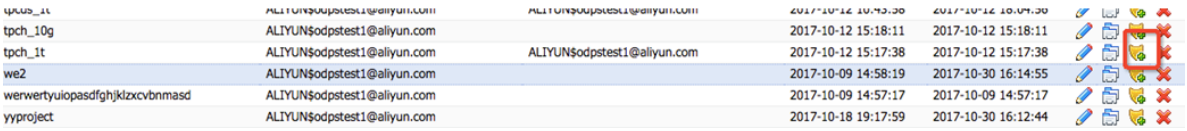
Project级别IP白名单配置

如果某一个project不在project group中时，则可以单独进行project级别白名单配置。

具体的配置方式如下所示。

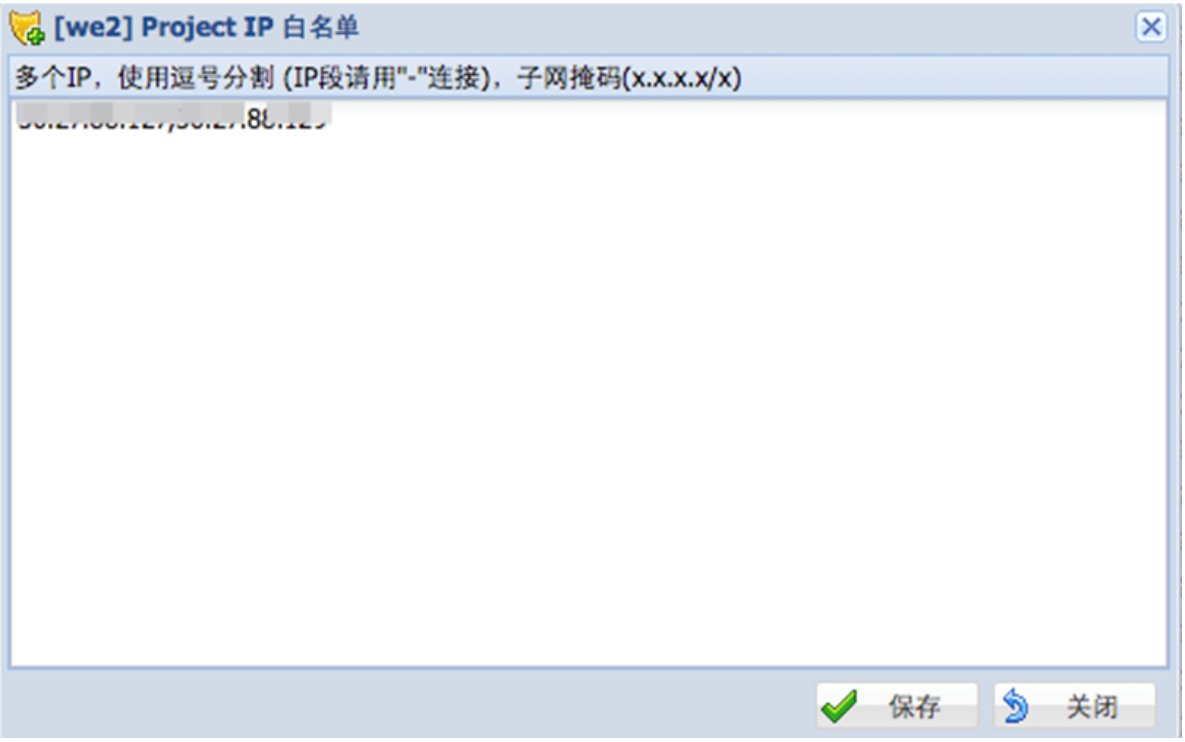
1. 在`AdminConsole`中选择**ODPS配置** > **Project管理**，选中需要配置的project，单击最右侧的IP白名单设置图标。

图 4-4: Project级别IP白名单配置1



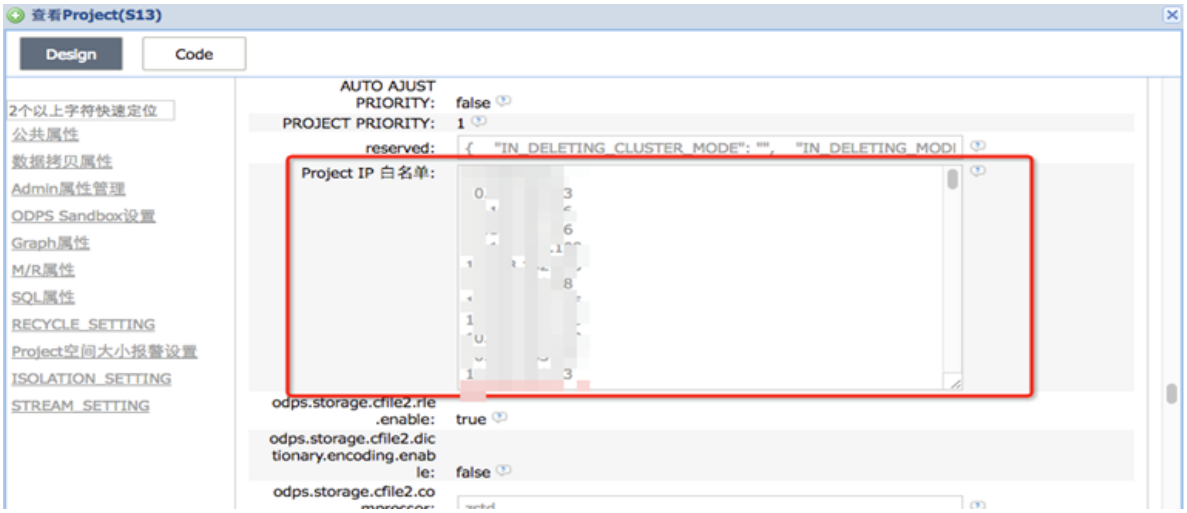
2. 在弹出的配置框中完成相关配置后，单击**保存**。

图 4-5: Project级别IP白名单配置2



3. 配置完成后可以到project属性配置中查看配置结果。

图 4-6: Project级别IP白名单配置3



说明：

Project owner也可以使用SetProject命令设置project的属性，如：setproject odps.security.ip.whitelist=“IP列表以逗号分隔”。

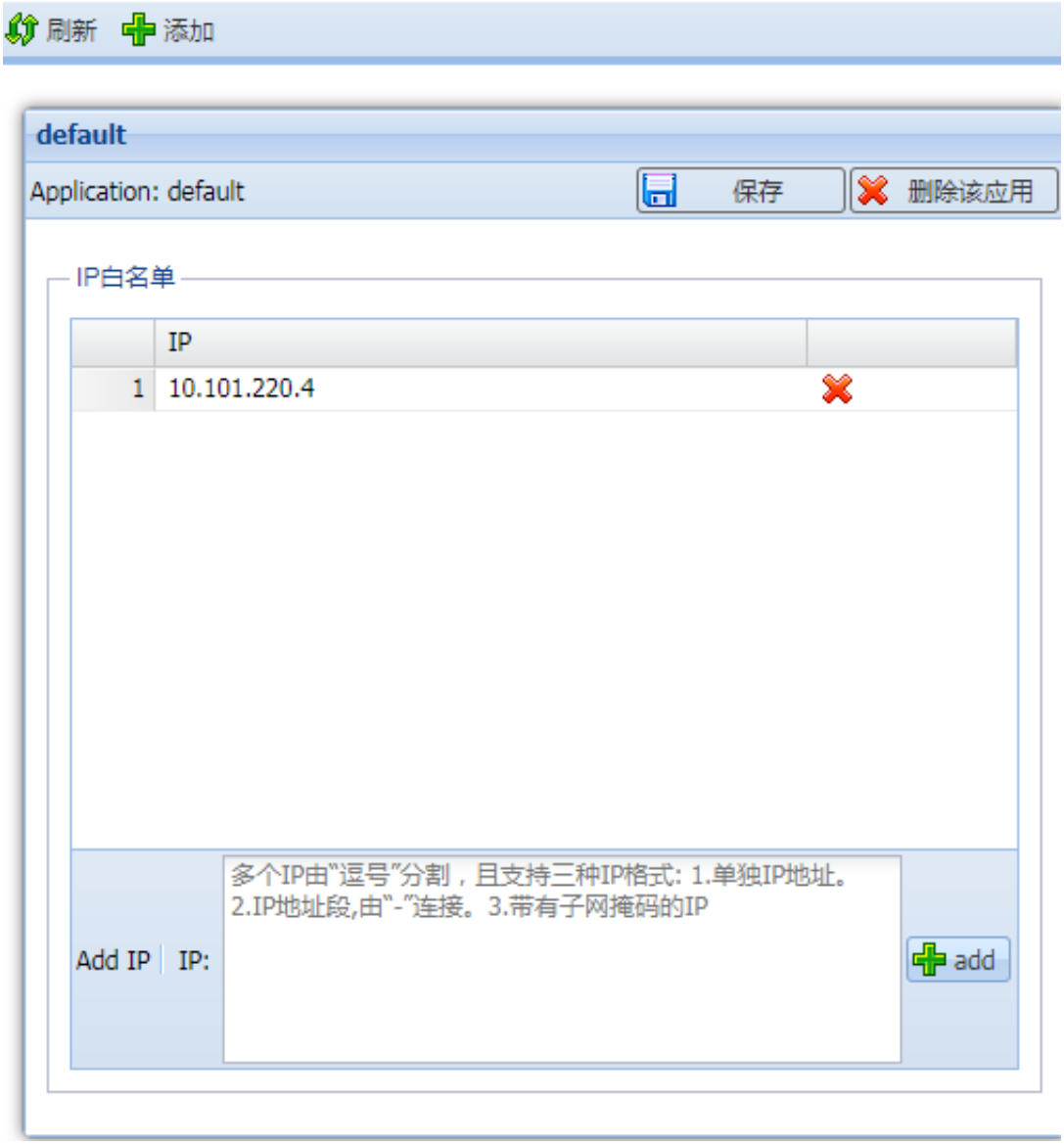
系统级别IP白名单配置

一些其他需要访问MaxCompute服务实例中所有project的其他上层业务系统（如Dataworks系统）IP发生变化的时候，如果没有全局性IP白名单配置，需要找到所有设置白名单的project列表一个个进行新IP的修改配置，非常容易出错。为此MaxCompute实现了系统级别IP白名单功能，系统级别IP白名单是MaxCompute实例服务级全局性配置。配置系统级别白名单是按照应用进行分类的。

具体的配置方式如下所示。

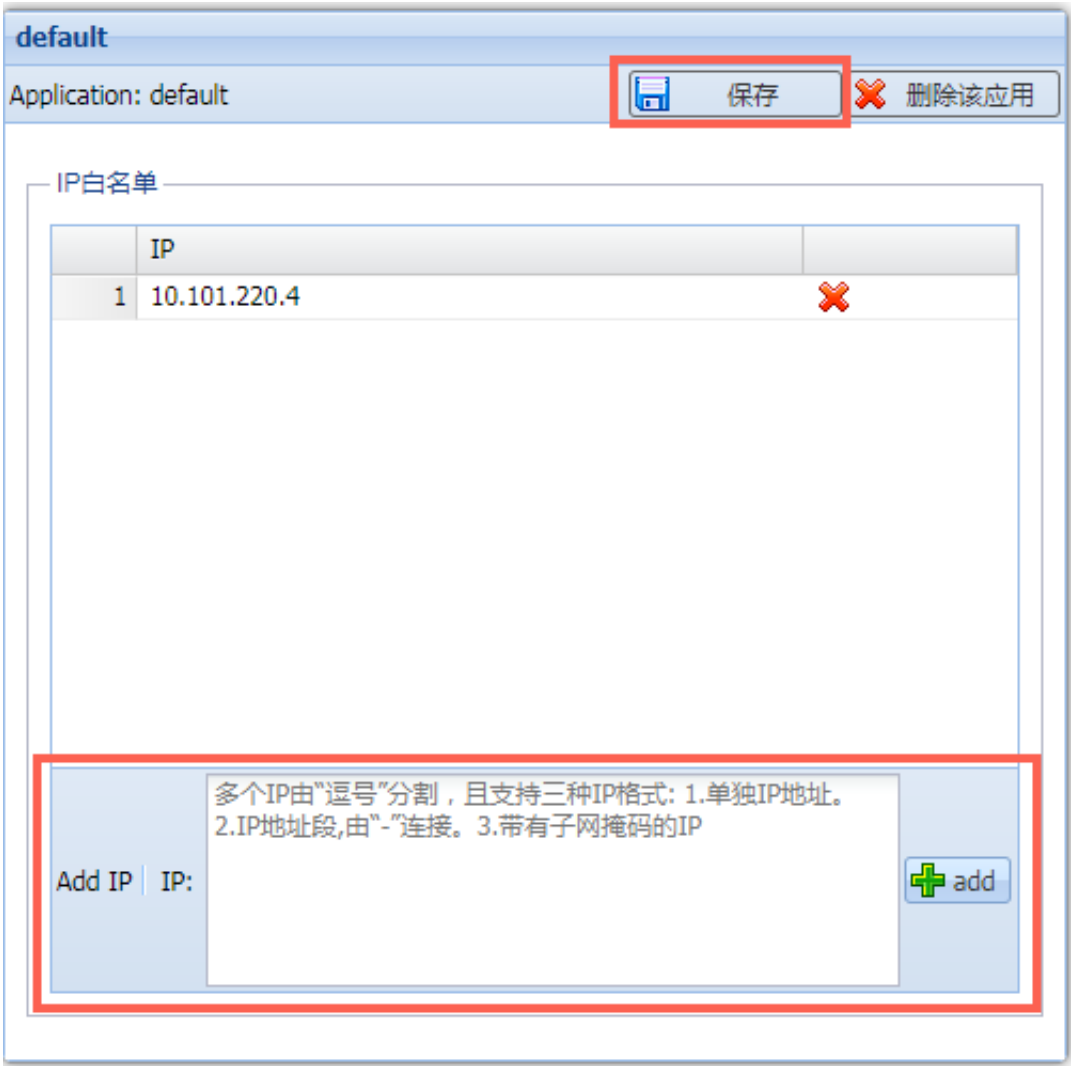
1. 在`AdminConsole`中选择**ODPS配置** > **系统级白名单管理**，默认打开配置框。

图 4-7: 系统级别IP白名单配置1



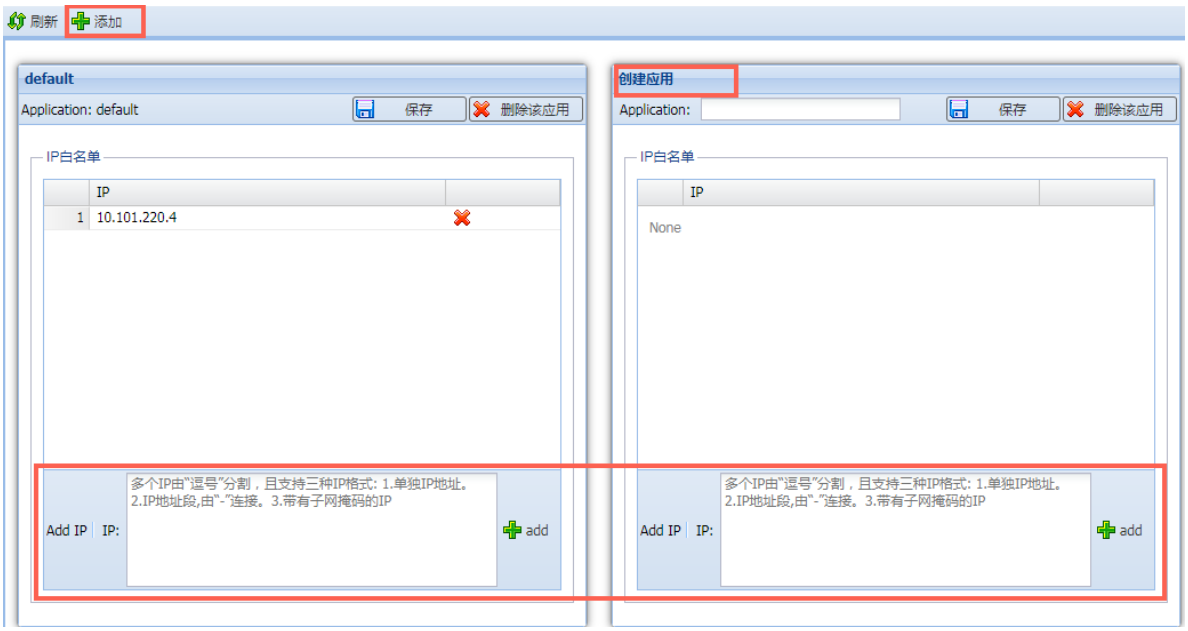
2. 在打开的配置框中完成相关配置后，单击**保存**。

图 4-8: 系统级别IP白名单配置2



3. 当前应用配置完成后，可以通过单击**添加**，继续配置新建应用的IP白名单。

图 4-9: 系统级别IP白名单配置3



相关注意事项

1. AdminConsole的专有云地址为：*http:///{odps_ag}:9090*，即odps ag的9090端口。
2. 首次设置白名单时一定注意需要设置正确的的白名单且包含本机IP地址，否则设置生效后本机IP地址不在白名单列表也会被限制不能访问。一旦设置错了之后，需要系统管理员从管理系统如AdminConsole中进行配置更改。
3. 给project或者project group设置完白名单之后，白名单之外的IP地址将无法访问受影响project，一些公用系统（如base）如果也需要访问该project，则需要设置base所在机器IP地址到白名单列表中。
4. 出于信息安全的考虑，即使IP白名单允许访问，用户也可以通过policy限制服务，这是另外一个层次更细粒度的访问控制。
5. 如果通过代理服务器访问MaxCompute服务， 需要添加到IP白名单的为最后一跳的代理服务器IP地址。

影响与效果

1. 配置之前MaxCompute服务针对访问项目空间的机器IP地址没有限制。
2. 配置之后，满足配置规则的IP地址及IP地址段才能访问该项目空间。在原有AccessKey ID及AccessKey Secret认证基础上叠加了IP规则的检查。

3. 一些公共系统，如Base，Datax，DPC系统原来需要访问到MaxCompute服务项目空间的，如果需要访问某一个项目空间，也需要找到这些服务部署机器的IP地址添加到IP白名单中。

4.5.7 MaxCompute支持VPC

大数据计算服务（MaxCompute）作为阿里云开发的海量数据处理平台，在安全性方面需要满足安全隔离规范的要求。因此，MaxCompute团队增加了MaxCompute对专有网络（VPC）的支持，为MaxCompute配置使用限制，即MaxCompute VPC的限制。

目前MaxCompute支持VPC的具体情况如下所示：

- 经典网络/VPC网络/Internet网络三网隔离，只能访问各自对应的end point及VIP。
- 经典网络能够访问所有project 。
- 没有配置VPC ID及IP白名单的project可以被三种网络中请求通过的相应域名访问，没有限制。
- 配置了VPC_ID的project只能被对应的VPC访问。
- 配置了IP白名单的project只能被对应的机器访问 。
- 对于加了代理的访问请求，判断为最后一跳代理IP及VPC ID为准 。

4.6 分析型数据库AnalyticDB

4.6.1 安全隔离

AnalyticDB以数据库作为租户隔离的基本单元，数据库创建者的云账号为数据库的Owner。未经数据库创建者授权，任何其他云账号不能访问该数据库的数据。用户的数据库在自己独享的进程级别实例上运行，从进程级别实现了数据库的隔离。

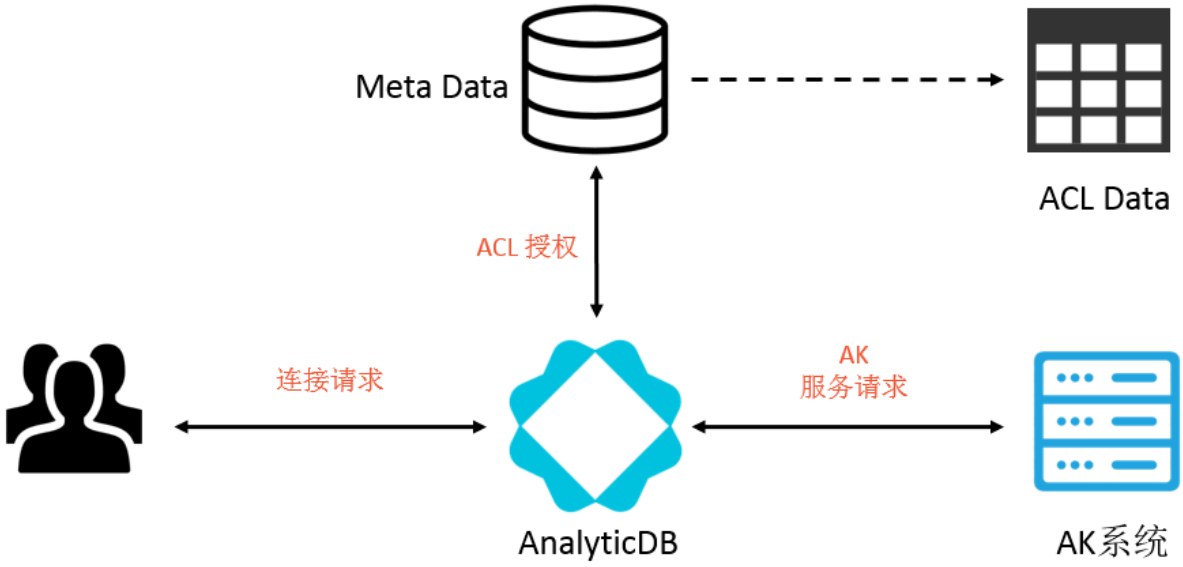
AnalyticDB集群中的每个数据库均采用多租户机制，每个数据库都有完全独立的服务进程。多租户机制对每个数据库的物理资源进行隔离（包括CPU、内存、存储空间），不允许跨数据库的访问。

AnalyticDB可以按数据库进行版本管理、资源扩容/缩容、数据库服务启动/停止。

从用户的访问权限认证角度来讲，每个数据库都有独立的AccessKey。

4.6.2 鉴权认证

AnalyticDB的身份验证和权限控制如下：



4.6.2.1 身份验证

AnalyticDB提供基于MySQL协议身份认证体系，支持类似MySQL的用户名/密码的身份认证机制。

作为阿里云产品栈产品，AnalyticDB使用阿里云AK（AccessKey）系统和机制实现身份认证。用户通过注册AK账户并进行登录，使用访问Key通过JDBC/ODBC连接数据库服务，通过AK服务请求实现身份认证。

用户可以在云控制台中自行创建AccessKey。AccessKey由AccessKeyId和AccessKeySecret组成，其中AccessKeyId是公开的，用于标识用户身份（相当于用户名）；AccessKeySecret是私密的，用于用户身份的鉴别（相当于密码）。

主账号和子账号均需要使用对应的AccessKey ID和Access Key Secret来访问AnalyticDB数据库。

4.6.2.2 权限控制

AnalyticDB支持基于数据库表的层级权限管理模型，提供类似MySQL的访问控制列表ACL（Access Control List）授权模式。与MySQL不同的是，AnalyticDB不支持针对用户在host上授权。

一个ACL授权由被授权的用户、授权对象和授予的对象权限组成。ACL数据存储在AnalyticDB的元数据系统中，元数据系统使用RDS持久化存储，同时元数据通过AnalyticDB的缓存进行DML/DDL语句授权管理的加速。

用户连接到AnalyticDB后，AnalyticDB通过ACL的元数据控制用户对数据库对象的操作权限，例如：用户对Table（表）、Column（列）等的SELECT、INSERT、DELETE、CREATE、SHOW、DROP、ALTER、DESCRIBE、LOAD DATA、DUMP DATA操作权限。

AnalyticDB中的授权对象如下：

- Database（库）：即 db_name.* 或 *（默认数据库），指定数据库或数据库上所有表/表组。
- TableGroup（表组）：即 db_name.table_group_name 或 table_group_name ，指定特定表组。
- Table（表）：即 db_name.table_name 或 table_name ，指定特定表。
- Column（列）：语法上由 column_list 和 Table 组成，指定表的特定列。

4.6.2.3 RAM和STS支持

AnalyticDB支持RAM（Resource Access Management）鉴权，不支持STS（Security Token Service）鉴权。

RAM 是阿里云提供的资源访问控制服务。通过RAM，主账号可以创建出子账号，子账号从属于主账号，所有资源都属于主账号，主账号可以将所属资源的访问权限授予给子账号。

4.6.3 数据安全

多租户

AnalyticDB提供多租户机制，不同数据库间通过CPU、内存、磁盘空间、网络带宽资源的完全隔离实现数据的隔离。

数据可靠性

AnalyticDB的全量数据保存在飞天操作系统的盘古分布式文件系统中，支持采用三副本或纠删码EC（Erasure Code）方式存储，提供数据持久化的高可靠性保证。实时表数据的DML语句（INSERT/DELETE）操作提交成功后，同步保存到盘古分布式文件系统中；对于批量表，数据加载时也全量写入到盘古分布式文件系统。

数据一致性

对于实时表的数据更新操作（INSERT/DELETE），AnalyticDB采用多版本并发控制MVCC（Multi-Version Concurrency Control）机制进行存储，以保证并发数据更新操作时查询所见数据为发起查询时的数据版本。

说明：
对于更新过的历史版本，如果不需要再查询引用，其空间将可定期清理。

4.6.4 日志审计

AnalyticDB支持开启审计日志，开启审计日志后可以记录所有SQL操作记录信息，包括：

- 查询发生时间。
- 客户端IP地址。
- 所执行SQL语句。
- 通过SQL语句可回看客户查看的数据信息。

审计日志格式示例如下：

[2017-10-10 13:37:57,351] INFO [pool-31-thread-22] c.a.c.a.f.l.AccessLog.info - Client=127.0.0.1 Total_time=1044 Exec_time=1043 Queue_time=1 - [2017-10-10 13:37:56 308] 1 SQL Statement \; process=20171010133756010003163108099998380 42\;CLUSTER=ayads-bjyz

4.6.5 VPC支持

AnalyticDB支持专有网络VPC（Virtual Private Cloud）功能，默认使用Single Tunnel方式，也可以通过配置切换到Any Tunnel方式。

专有网络VPC可以帮助您基于阿里云构建出一个隔离的网络环境。您可以完全掌控自己的虚拟网络，包括选择自有IP地址范围、配置路由表和网关等。此外您也可以通过专线 、VPN等连接方式将VPC与传统数据中心组成一个按需定制的网络环境，实现应用的平滑迁移上云。

- **Single Tunnel模式**：默认的VPC方式，仅支持在指定的VPC环境中访问并使用AnalyticDB。Single Tunnel模式可以实现不同VPC之间的网络隔离。
- **Any Tunnel模式**：通过修改配置，可从Single Tunnel模式切换到Any Tunnel模式。配置修改后在下一次创建数据库时生效，您也可通过修改元数据并重启FrontNode来使配置变更生效。Any Tunnel模式下，您可在任意的VPC环境中访问并使用AnalyticDB。Any Tunnel模式无法实现VPC之间的网络隔离。

4.7 关系网络分析I+

4.7.1 安全隔离

I+针对用户的数据，进行了租户级的隔离，即不同租户相互之间不能查询到数据。租户只能获取自己租户下的元数据配置，而不同的元数据对应不同的业务数据，所以同一租户只能查询到自己元数据对应的业务数据。

4.7.2 鉴权认证

4.7.2.1 身份验证

I+关系网络分析目前在专有云支持两种身份验证：

- I+自己的身份验证：通过I+用户系统创建的用户密码登录，I+用户系统的密码经过MD5加密，并且在网络传输上也经过加密，有效防止了密码泄漏的情况。
- 对接的外部系统身份验证：对接客户的用户系统，该种方式的用户安全由外部系统承担。

4.7.2.2 权限控制

I+关系网络分析产品，所有功能都有权限控制，可以根据不同的用户权限对产品功能模块、数据行列进行管控。

4.7.3 数据安全

I+关系网络分析采用分布式集群部署，管理节点和计算节点分离，能有效防止系统的单点故障，并且集群之间采用分布式缓存同步，有效防止了系统在故障转移时出现的数据丢失。

4.7.4 传输加密

I+关系网络分析产品以HTTPS协议提供web服务。HTTPS协议是一种安全可靠的数据传输协议，能有效防止数据在网络上传输带来的安全问题。

4.7.5 日志审计

I+关系网络分析产品中所有的用户请求均记录日志，作为审计或者特定行为分析使用。日志中包含用户、IP、操作内容、操作状态等信息。

4.7.6 系统安全

4.7.6.1 漏洞扫描机制

I+关系网络分析产品在发布前，已经经过专有云安全漏洞扫描，并且通过安全扫描，扫描内容包括：

- 系统安全扫描：I+关系网络分析产品发布的操作系统的安全扫描。
- 中间件依赖扫描：I+关系网络分析产品使用到的中间件。
- 代码漏洞扫描：I+关系网络分析产品自己的代码，以及依赖的第三方开源框架。

4.7.6.2 安全漏洞更新修复方案

根据阿里云安全部分、专有云安全测试、以及其他途径获取的安全漏洞，I+产品研发团队将根据安全漏洞的影响程度，进行紧急版本更新或者版本迭代更新，而且无论哪种更新都会保障更新流程符合阿里云安全生产管理规范。

4.7.6.3 系统防御机制

I+关系网络分析产品是基于阿里云专有云环境发布的，I+关系网络分析产品的系统防御机制依赖于阿里云专有云系统的防御机制。

4.7.7 基础设施安全

I+关系网络分析产品是基于阿里云专有云环境发布的，基础设施的安全有阿里云专有云基础设施安全保障，可参见阿里云安全白皮书基础设施安全。

4.7.8 等保认证

I+关系网络分析产品已经对接了专有云V3.3的安全等保4级。从阿里云专有云天基环境获的安全证书包括：cacert.pem、privatecloud.pem、privatecloud_key.pem、privkey.pem。

I+关系网络分析在专有云V3.3版本以后通过https协议访问。

4.8 流计算StreamCompute

4.8.1 账号安全

流计算账号安全

流计算账号当前支持且仅支持阿里云账号体系（包括登录用户名+密码、签名密钥），这部分全部遵守阿里云现有安全体系，同时传输链路全部使用HTTPS协议，保证全链路的用户账户安全。

数据存储账号安全

流计算涉及到保存数据存储连接账号问题，我们提供基于RAM/STS方式，避免您因为账户信息丢失导致业务信息泄露。

4.8.2 业务安全

项目隔离安全

流计算对不同的项目进行了严格的项目权限区分，不同用户/项目之间是无法进行访问、操作，包括项目下属的所有子产品实体均无法操作。

项目级别的资源隔离能够保证不同用户的资源使用情况相互之间不相互干扰影响，例如一个用户任务在运行期间随着数据量的突增提升了其作业CPU使用。阿里云流计算在底层使用虚拟化技术进行资源隔离，保证该用户的作业CPU使用率增加不会影响到其他用户作业的CPU使用情况。

业务流程安全

流计算对于流式计算开发进行了严格的流程定义，区分了数据开发和数据运维，在尽可能不影响用户使用体验基础上，保证了整体业务流程的完整和安全性。

- **提供代码版本**

  支持代码版本回滚和对比，方便您对代码进行追溯、比对、排错。

- **提供IDE单机调试容器**

  避免代码线下运行影响线上真实数据。您可以对输入表、维表、输出表自行构造数据，以避免线下任务调试对于线上生产任务影响。

- **提供发布流程**

  避免线下代码改动直接影响生产运行。用户调试完成后，通过上线任务将作业提交到数据运维系统。此时正在运行的流计算任务并不直接使用新代码运行，而需要您经过人工确认后将运行任务停止并使用新代码启动，从流程上保证发布的严谨性。

4.8.3 数据安全

数据安全分为流计算系统数据安全和业务数据安全。

系统数据安全

流计算系统数据安全交由系统本身安全保证，流计算为系统安全做了诸多工作。

- 访问链路全部HTTPS化，保证传输链路的安全。
- 数据存储连接信息使用AES高强度加密方式，保证敏感信息不泄露。
- 全面且深入的攻击测试，阿里云安全团队为流计算保驾护航。

业务数据安全

流计算本身不负责存储用户的业务数据，具体业务数据安全交由不同的阿里云存储系统保证，详情请参见不同的数据存储的安全模型以及最佳安全实践。