

URLOS使用手册 v 0.4.0

URLOS使用手册 v 0.4.0

安装与常见问题

简介

概述

产品特点

相关概念

环境要求

硬件要求

系统要求

安装URLOS

升级URLOS

授权码绑定

常见问题

不能访问URLOS面板

忘记URLOS主控端密码

不能安装URLOS或安装失败

首页管理

服务概览

集群/节点/应用统计

软件运行环境

建站流程

系统管理

系统工具

系统设置

资源管理

主机集群

基本信息

本地存储

共享存储

节点内核优化

容器DNS服务器

主机节点

基本信息

SSH登录

系统优化

挂载硬盘

其他设置

容器网络

安装应用

应用市场

安装数据库服务

基本信息

设置

快照与备份

资源限制

其他设置

安装网站环境

基本信息

网站

数据库

快照与备份

攻击防护

上传下载

安全证书

PHP配置

Nginx配置

资源限制

其他设置

安装CDN网站加速

购买应用

基本信息

网站

源站设置

目录缓存

文件缓存

安全证书

Nginx配置

CDN调优

资源限制

其他设置

服务管理

网站服务

服务日志

快照和备份列表

手动快照备份

容器信息

查看活动容器数

创建相似服务

停止并删除服务

数据存储

管理数据库

添加数据库

修改数据库

删除

测试系统

数据缓存

所有服务

监控分析

网站统计

开启监控分析功能

微信查看主机监控

网站日志清理

日志管理

任务计划

系统日志

###快速入门###

5分钟从0开始创建网站

外部连接MySQL数据库

通过命令行进入容器和退出容器

制作自己的Docker镜像

安装与常见问题

简介

概述

URLOS是一个Docker管理面板，它把服务器端软件应用的安装行为简化到极致，堪称服务器端的应用宝，具有 [集群管理](#)、[自动故障转移](#)、[自动负载均衡](#) 等高级功能，可轻易搭建7*24小时在线的网站运行环境。

借助URLOS可轻松让网站在拥有多个主机的集群上运行，单机故障不会导致网站停止。

演示网址：<http://demo.urlos.com:9968> (用户名和密码使用默认的urlos，填写验证码后即可登录)



产品特点

安全

- 基于Docker容器技术，应用均运行在具有隔离功能的容器中，如果某个应用出现问题（包括假死、真死、漏洞、后门等问题），并不会影响其它应用的正常运行，大大提高安全性。
- 默认情况下以非特权用户在容器中隔离运行应用，应用出现安全漏洞或后门并不会影响母机安全，就如默认情况下手机上的应用并不会威胁手机操作系统的安全一样。

稳定

- 最低只用3台服务器即可构建7*24小时在线的网站集群环境，如果其中1台服务器出故障，并不会影响网站业务的正常运行（自动发现故障+自动故障转移+自动负载均衡），添加新的服务器到服务器集群即可全

自动修复!

- 针对某网站进行CPU和内存资源的限制, 解决“由于1个网站不稳定, 导致同1台服务器上的100个网站都打不开”的棘手问题。

包容

- 可在同1台服务器中运行100个不同的网站环境, 互不冲突, 包括Nodejs, Golang, Python, Lua, .NET, PHP4X, PHP5.2, PHP5.3, PHP5.5, PHP5.6, PHP7.0, PHP7.1, PHP7.2等100种以上的网站运行环境。
- 可在同1台服务器中运行100个不同的操作系统, 互不冲突, 包括CentOS, Debian8X, Debian9X, Ubuntu14X, Ubuntu16X, Ubuntu17X, Alpine等Linux操作系统。

易用

- 在网页控制面板上选择并安装应用软件, 简易度接近手机软件的安装: 选择应用->安装应用->自动部署应用, 根本无需考虑环境兼容性问题——解决环境兼容性问题开发人员的事, 不是您老的事!
- 只在安装URLOS时需要输入命令行, 在使用URLOS的过程中, 可远离命令行——所有需要SSH才能完成的维护工作, 都有改良空间, 因为不够自动化、维护成本高、出错率高等等。

体贴

- 我们希望管理服务器就像管理自己的手机一样简单, 安装和管理服务器端应用就像安装和管理手机应用一样傻瓜化, 没有教程也不需要教程, 我们相信那一天很快就会来临!
- 我们期盼着有一天维护人员可以彻底解放和释放自己, 有更多时间和精力与亲朋好友一起, 在蓝天白云下、高山流水间...尽情地欢愉、跳跃!

相关概念

- **应用**: 应用存放于URLOS的应用市场, 相当于手机上的APP;
- **镜像**: 用于存放应用安装包的存储介质, 可理解安装软件的光盘;
- **插件**: 每个应用可选择绑定一个插件, 在安装和启动应用时执行一些个性化或自动化操作, 还能用户提交的数据进行检查等;
- **服务**: 将应用安装并运行后, 便可提供服务, 如用于存储数据的数据库服务, 用于提供WEB访问的网站服务等, 基于一个应用可安装并运行多个服务;
- **节点**: 装有Linux系统的服务器或云主机即为节点;
- **集群**: 集群由多个节点组成, 具有可伸缩性、高可用性等特点, 单个节点出现故障一般不会影响集群内的业务;
- **容器**: 用于运行应用的“盒子”, 运行在集群的节点上, 当某个节点出故障时, 可以迅速转移到集群中的另一节点并启动应用, 从而提供几乎不间断的服务;

环境要求

硬件要求

最低硬件配置: 1核CPU, 1G内存 (1+1) 提示: 如果你的应用较多, 而主机节点的硬件配置较低, 建议在部署节点时开通虚拟内存;

系统要求

推荐安装系统: Ubuntu-16.04、Ubuntu-18.04、CentOS7.X、Debian8X、Debian9X的64位的 [纯净的操作系统](#);

安装URLOS

1. 登陆操作系统：使用ssh客户端工具登录系统，如果登录用户不是root，则执行 `sudo -s` 命令切换到root用户；
2. 下载安装脚本并执行安装命令：

```
curl -SO https://www.urlos.com/install && chmod 544 install && ./install
```

或

```
curl -O https://www.urlos.com/iu && sh iu
```

注：如果系统中没有curl工具，则需先执行“`apt-get install -y curl || yum -y install curl`”安装curl

3. 选择服务器所在区域的编号：服务器在中国选1，如果1不能安装则选2，如果服务器在海外则选3（如下载过程比较缓慢，请换一个编号），然后按回车；
4. 选择Docker数据的存储目录：/data/docker选1，/home/docker选2，/var/lib/docker选3，然后按回车；
5. 安装成功并重启服务器：安装完成后可能会自动重启服务器，重启过程一般需要1分钟左右；
6. 访问并使用URLOS：请在防火墙（或安全组）中开放9966,9967,9968,9969端口（进站规则）和8866端口（出站规则），然后使用浏览器访问“https://您的服务器IP:9966”或“http://您的服务器IP:9968”访问URLOS；用户名：urlos，密码：urlos

注：建议使用https协议登录URLOS，相对更安全！

升级URLOS

1. 登陆操作系统：使用ssh客户端工具登录系统，如果登录用户不是root，则执行 `sudo -s` 命令切换到root用户。
2. 下载升级脚本并执行升级命令：

```
curl -SO https://www.urlos.com/update && chmod 544 update && ./update
```

注：如果升级时无法停止MySQL程序，请使用强制升级：

```
curl -SO https://www.urlos.com/update && chmod 544 update && ./update latest force
```

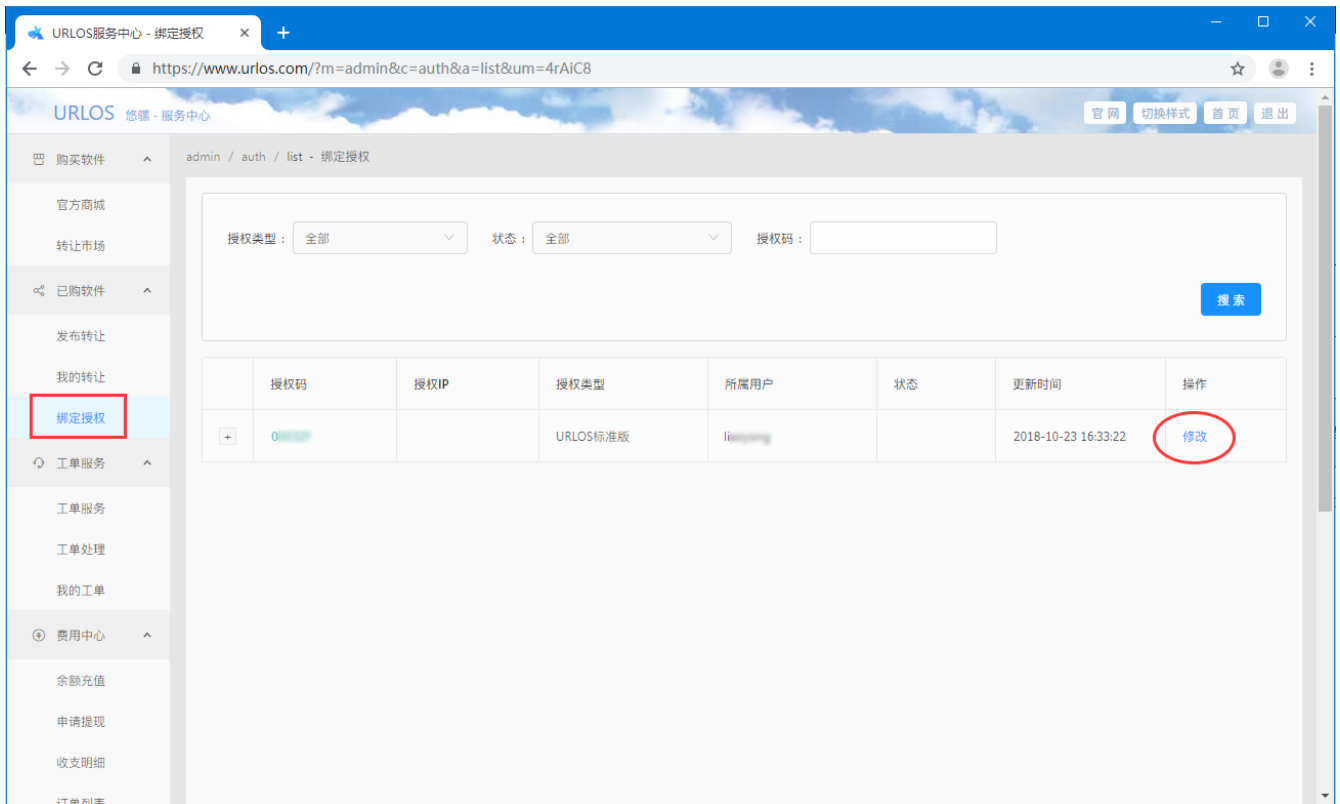
3. 选择服务器所在区域。
4. 选择Docker数据的存储目录：必须设置成旧的docker存储目录（如果忘记了，请执行 `cat /etc/docker/daemon.json` 查看！）
5. 访问并使用URLOS：安装完成后，请使用浏览器访问“https://您的服务器IP:9966”或“http://您的服务器IP:9968”访问UrIOS

提示：如果是从0.3.x升级到0.4.x版本，请在升级后登录URLOS [修改已安装的数据库服务的密码](#) 并 [暴力部署](#) 一次！

授权码绑定

URLOS授权码绑定的方法：

- 1、在服务中心绑定IP。登录服务中心(<https://www.urlos.com/>)，在左侧菜单中找到“[已购软件](#) > [绑定授权](#)”，在需要绑定的授权码后面点“[修改](#)”，在授权IP中填写服务器的 [外网IP](#) 地址。



2、修改URLOS配置文件。使用SSH工具连接服务器，输入以下命令来更新URLOS配置文件中的授权码信息：

```
docker exec urlos /usr/local/urlos/bin/common/tools.sh updateLicenseKey BBBBBB
```

其中 BBBBBB 代表要绑定的授权码。

3、在浏览器中访问URLOS管理平台，登录后查看授权版本是否已经更新。

常见问题

不能访问URLOS面板

问题原因：未放行服务器端口

处理方法：找到云主机服务商网站，登录服务商管理系统，放行 9966、9967、9968 端口。

忘记URLOS主控端密码

处理方法：通过SSH命令重新初始化密码。

操作步骤如下：

1. 请确保URLOS为0.2.28以上版本，如不是，请先升级；
2. 通过SSH登录URLOS所在的母机，执行 `sudo -s` 切换为root用户；
3. 执行下面的命令修改密码（其中的newPassword为你的新密码）：

```
docker exec urlos /usr/local/urlos/bin/common/tools.sh initAdminPassword newPassword
```

不能安装URLOS或安装失败

问题原因1：用户所使用的系统不符合URLOS安装要求；

处理方法：更换为符合URLOS需求的系统，如：Ubuntu-16.04、Ubuntu-18.04、CentOS7.X、Debian8X、Debian9X的64位的 [纯净的操作系统](#)。

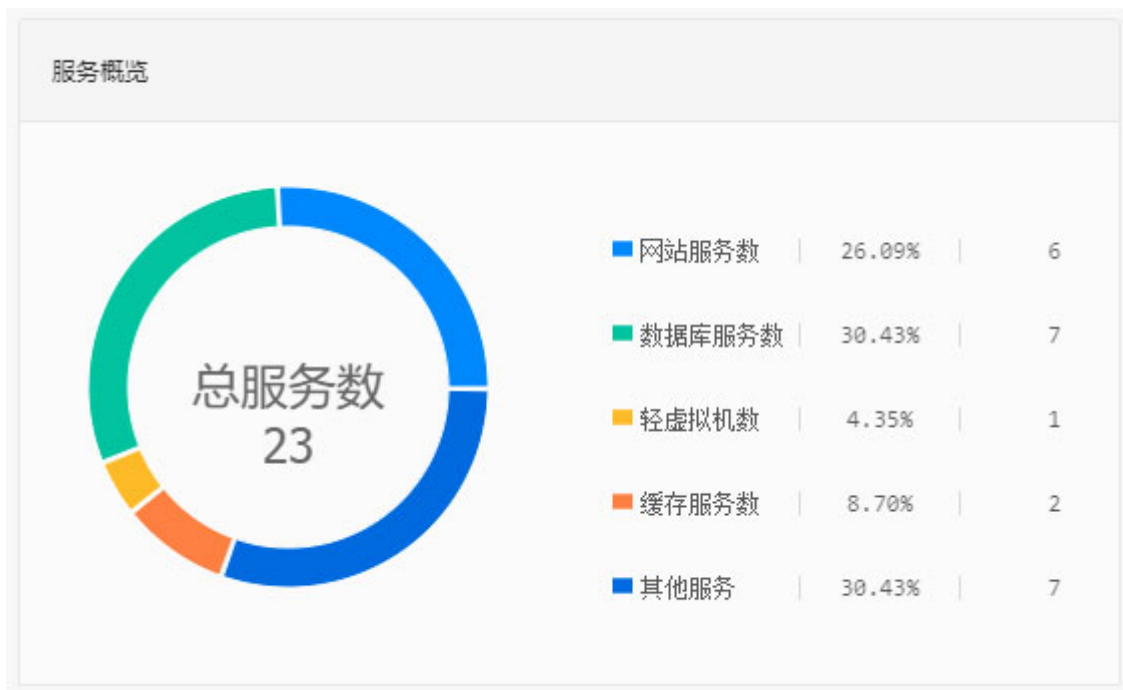
问题原因2：用户服务器访问无法正常访问安装镜像源；

处理方法：重新安装URLOS时，更换一下所在区域。

首页管理

服务概览

显示当前URLOS网络环境中运行的服务数量以及各自占比。



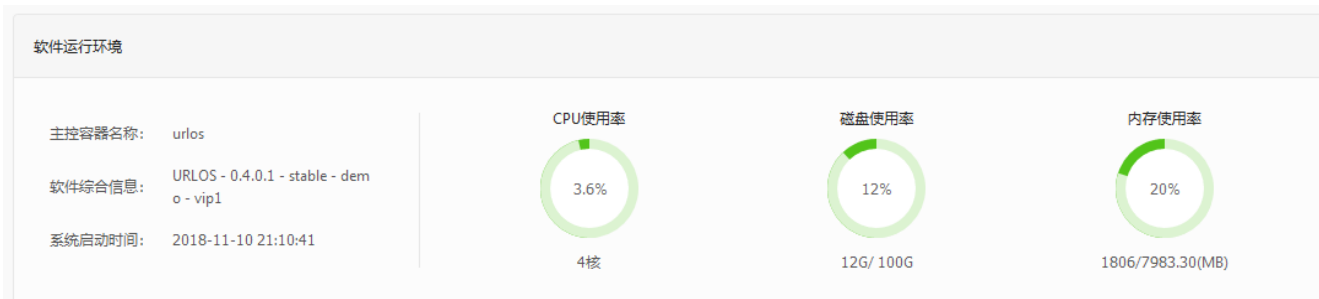
集群/节点/应用统计

显示当前URLOS网络环境中的集群数量、节点数量以及应用市场所承载的应用数量。



软件运行环境

显示主控容器名称、URLOS版本信息、服务器系统启动时间以及硬件资源使用情况。



建站流程

显示URLOS基本建站流程:

1. 添加集群和节点
2. 安装网站环境
3. 上传网站程序
4. 检查网站端口放行情况
5. 访问网站



系统管理

系统工具

系统工具，主要用于执行一些系统操作，如清除缓存并同步应用、强制部署所有服务、取消当前任务计划、取消所有任务计划、自动添加集群和本机节点、查看授权密钥、查看应用预共享密钥。

名称	标识	描述	操作
清除缓存并同步应用	flushCache	清除所有缓存文件并同步最新应用	执行
强制部署所有服务	forceDeployAllServices	一次性部署所有服务，一般是迁移和测试时使用	执行
取消当前任务计划	cancelCurrentCronProcess	取消当前正在执行的任务计划	执行
取消所有任务计划	cancelAllCronProcess	取消包括正在执行和等待执行的所有任务计划	执行
自动添加集群和本机节点	initLocalSwarmAndNode	创建一个集群并将urlos所在的节点加入到集群中	自动添加 手动添加集群
查看授权密钥	showLicenseKey	查看URLOS的授权密钥	执行
查看应用预共享密钥	showAppSharePassword	显示导入应用时的预共享密钥	执行

Version: 0.3.34-beta, Website: urlos.com, Runtime: 0.174s, MemUsage: 7.474m
广州市万岁云计算有限公司 版权所有

清除缓存并同步应用：清除所有缓存文件并同步最新应用。

强制部署所有应用：一次性部署所有服务，一般是迁移和测试时使用。

取消当前任务计划：取消当前正在执行的任务计划。

取消所有任务计划：取消包括正在执行和等待执行的所有任务计划。

自动添加集群和本机节点：创建一个集群并将urlos所在的节点加入到集群中。

查看授权密钥：查看URLOS的授权密钥。

查看应用预共享密钥：显示导入应用时的预共享密钥。

系统设置

系统设置，主要用于配置URLOS主控端的系统参数，常用配置如 [使用强密码验证](#)、[Docker镜像加速源](#)、[Linux软件镜像源](#)、[验证码类型](#)。

URLOS 让应用持续提供服务

应用市场 生成密码 清除缓存 诊断 官网 首页 换肤 退出

admin / setting / list - 系统设置列表

添加系统设置

名称	选项	选项值	所属用户	修改时间	操作
日志数据集中存储节点	logsNodeId		nobody	2019-02-17 09:56:54	修改
页面动画类型	animationType	fadeIn	nobody	2018-09-05 18:35:19	修改
使用强密码验证	forceStrongPasswords	1	nobody	2018-04-10 22:35:35	修改
容器退出时的等待时间	scriptsErrorExitWaitTime	60	nobody	2018-01-07 14:35:55	修改
Docker镜像加速源	dockerImageMirror		nobody	2019-02-19 11:35:00	修改
Linux软件镜像源	linuxSoftwareMirrors		nobody	2018-10-27 12:17:27	修改
通知检查间隔时间	notificationCheckTime	2	nobody	2017-12-16 21:43:59	修改
开启keeper调试模式	keeperDebug	0	nobody	2018-01-16 23:59:05	修改

使用强密码验证：此项开启时，创建数据库和SFTP密码必须包含数字、英文字母大小写、符号等字符。如需关闭强密码，则在选项值中填写“0”。

URLOS 让应用持续提供服务

应用市场 生成密码 清除缓存 诊断 官网 首页 换肤 退出

admin / setting / edit - 修改系统设置

系统设置管理

1. 基本信息

* 选项名: forceStrongPasswords

* 名称: 使用强密码验证

选项值: 1

* 所属用户: nobody

描述: 关闭则允许使用弱密码，建议不要在生产环境下开启
1: 开启; 0: 关闭

[提交](#)

Docker镜像加速源：用于切换Docker镜像下载源，如在安装应用过程中出现超时或者安装缓慢的情况，请在此切换其他镜像加速源。

目前可选择的加速源有：

```
registry.docker-cn.com
index.docker.io
registry.urlos.com:5000
?.mirror.aliyuncs.com
```



验证码类型：用于切换验证码显示类型，系统默认**牛人验证码**，机器难以识别，可有效防止暴力破解。如图：



系统提供3种验证码类型：

1. 牛人验证码
2. 素人验证码
4. 普通验证码

如需更换，则在选项值中填入相应数字。



资源管理

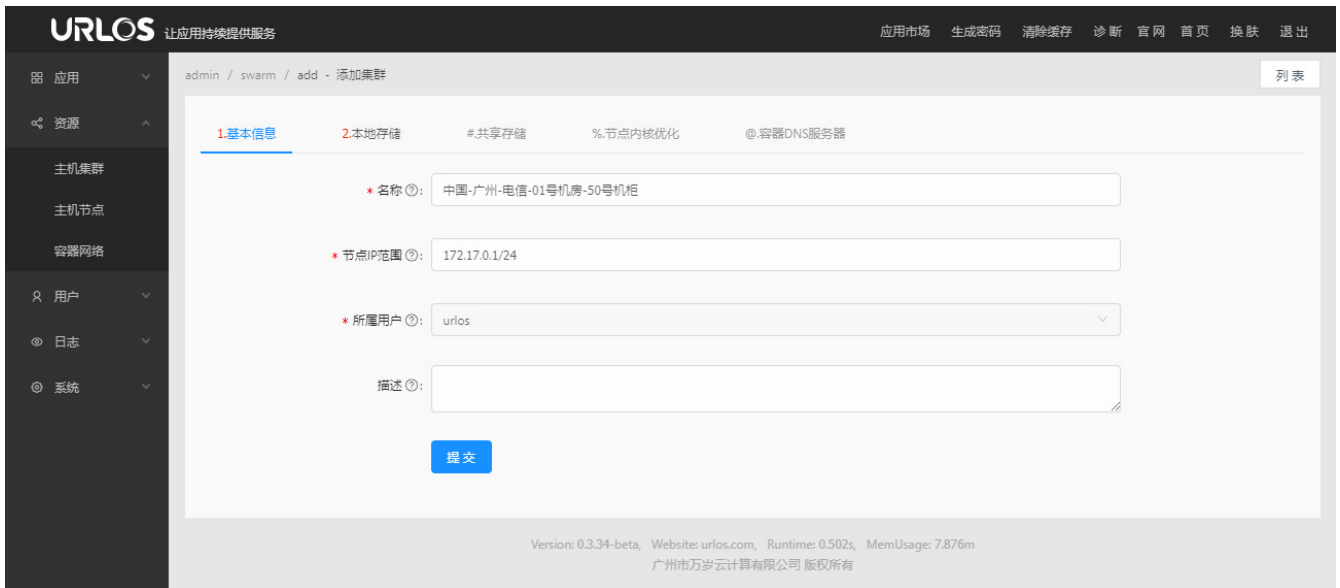
主机集群

打开 [资源](#) 菜单，选择 [主机集群](#)，然后在右边点击 [添加集群](#)，如下图：



基本信息

打开 [基本信息](#) 选卡：

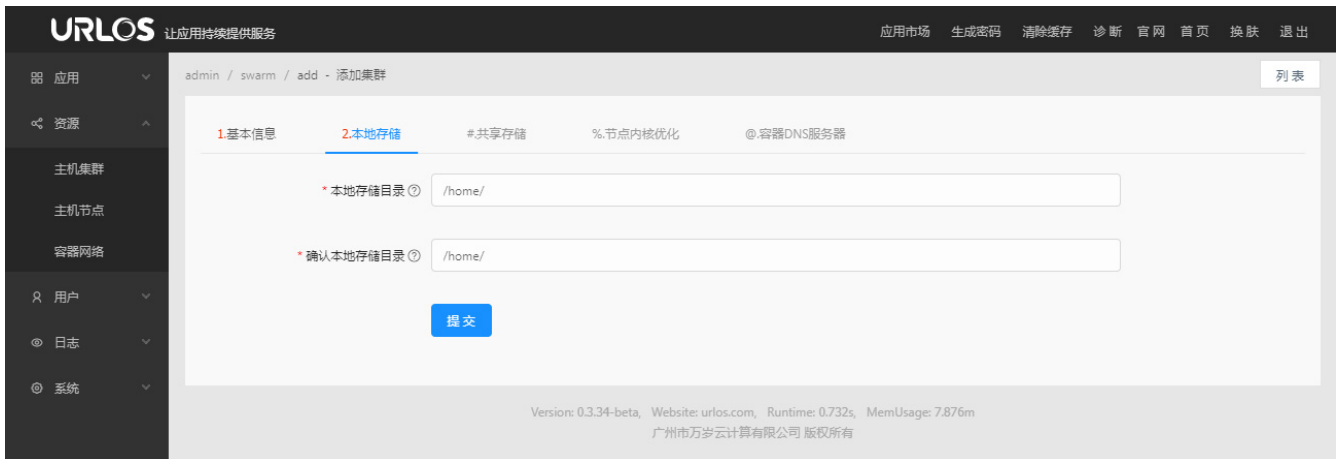


名称：填写当前集群的名称，如 [中国-广州-电信-01号机房-50号机柜](#)。

节点IP范围：填写当前集群的IP范围，如集群中有3个节点，分别为[192.168.0.1](#)、[192.168.0.2](#)、[192.168.0.3](#)，则填写“[192.168.0.1-3](#)”。

本地存储

打开 [本地存储](#) 选卡：



本地存储目录：请定义本集群内节点的本地存储目录（暂时仅支持本地存储），即用于存储数据的目录，通常为数据盘的的挂载目录，如：[/data/](#)、[/home/](#)

共享存储

打开 [共享存储](#) 选卡：



本地挂载目录: 用于挂载NFS(或NAS)的本地目录, 即映射到本地节点的文件夹, 一般可设置为 `/nfs-data/`。

NFS主机: 请填写NFS服务器的IP或主机名, 如 `9d08f490cc-avt64.cn-zhangjiakou.nas.aliyuncs.com`。

NFS共享目录: 服务器端共享的文件夹路径, 一般是NFS服务器端的NFS存储目录的路径, 如: `/home/nfsDir`

NFS版本: 请选择NFS对应版本。

节点内核优化

打开 [节点内核优化](#) 选卡:



一般情况下, 请不要修改默认参数, 以免不合理的设置造成节点性能下降。

容器DNS服务器

打开 [容器DNS服务器](#) 选卡:



一般情况下，无特殊需求则不必设置该选项。

主机节点

基本信息

打开 [基本信息](#) 选卡：



主机名称：请输入主机名称，由字母、数字、中划线和点组成，如：IP100。注：仅有开发和维护模式下才可以修改主机名称。

IP地址：请输入IP地址，如：192.168.1.188。

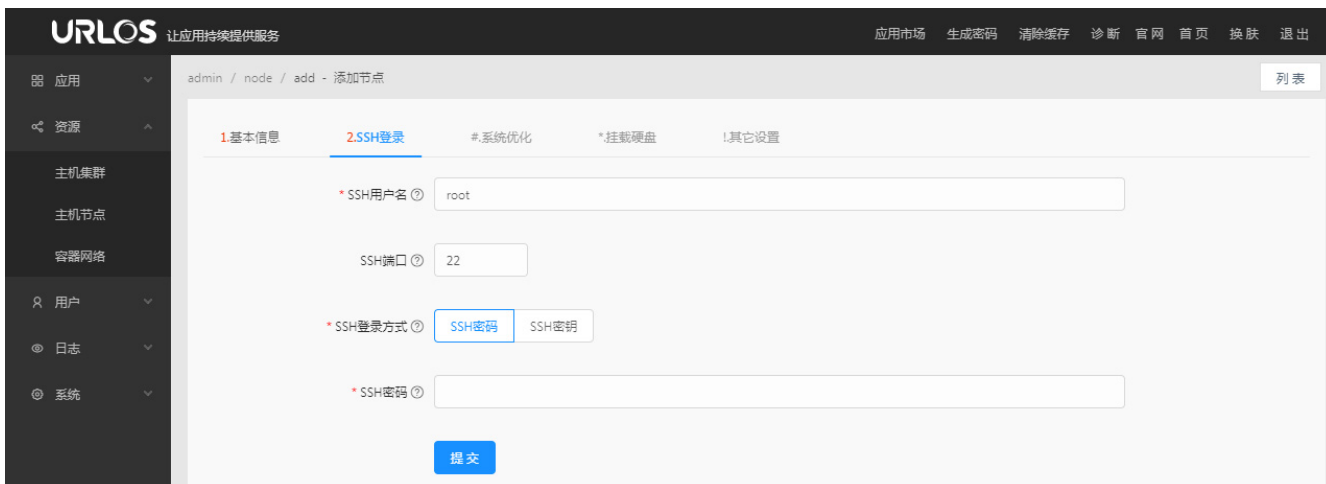
所在集群：选择当前节点所在集群。

节点类型： [管理节点](#) 用于管理集群节点，当管理节点数量大于等于3个时，集群便有了自动修复功能，当集群内某个节点宕机时，基于非单机应用构建的服务可实现自动迁移，业务几乎不中断； [工作节点](#) 仅仅被用于运行容器应用，其不具备管理功能。

状态：控制节点开启或关闭退出集群。

SSH登录

打开 [SSH登录](#) 选卡：



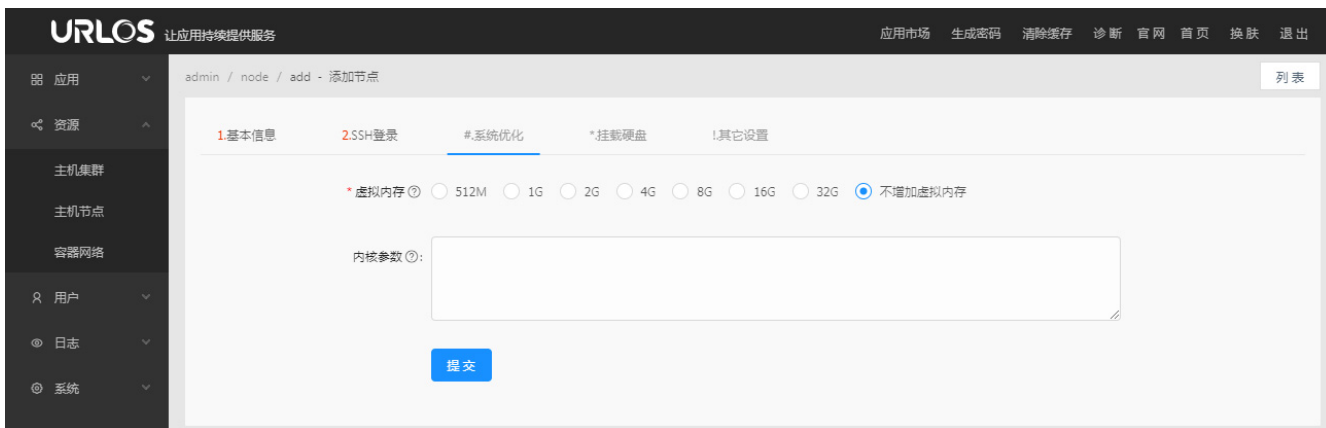
SSH用户名: 请输入SSH用户名, 除非你的Linux系统可以免密码切换(执行 `su` 或 `sodu -s` 命令时不需输入密码)到root用户, 否则必须使用root用户。

SSH端口: 请输入SSH连接端口, 如: `22`。

SSH登录方式: 支持SSH密码或SSH密钥登录, 请选择其一进行登录。

系统优化

打开 [系统优化](#) 选卡:

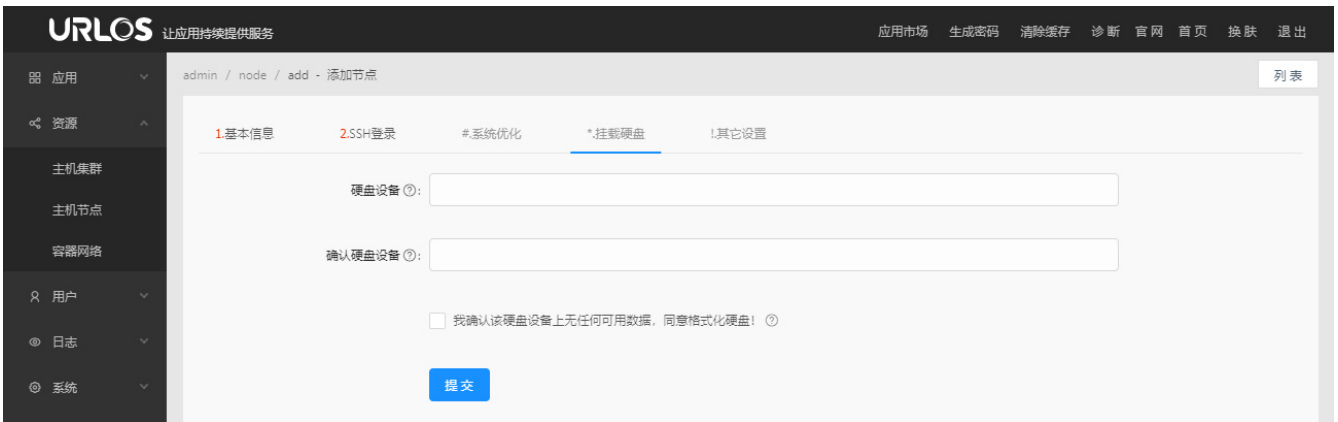


虚拟内存: 当节点的内存不够用时, 可能会导致无法部署服务, 开启虚拟内存可临时解决此问题, 设置后不可修改 (如果条件允许, 请尽量不要使用虚拟内存, 因为虚拟内存会导致系统性能下降), [建议1G物理内存配置1G虚拟内存, 2G物理内存配置2G虚拟内存, 依此类推](#)。

内存参数: 用于调整系统Sysctl的值, 内容为json格式, 如: `{"net.ipv4.tcp_mem": 10407, "net.ipv6.conf.default.mtu": 1280}`, 注意: 此处配置的内核参数相对集群中的内核参数具有优先级。

挂载硬盘

打开 [挂载硬盘](#) 选卡:



硬盘设备: 请添加硬盘设备, 如: `/dev/sdb`

勾选将擦除硬盘的现有数据, 并且将硬盘挂载到存储目录下的mounts目录上 (一般为 `/data/mounts` 或 `/home/mounts`)

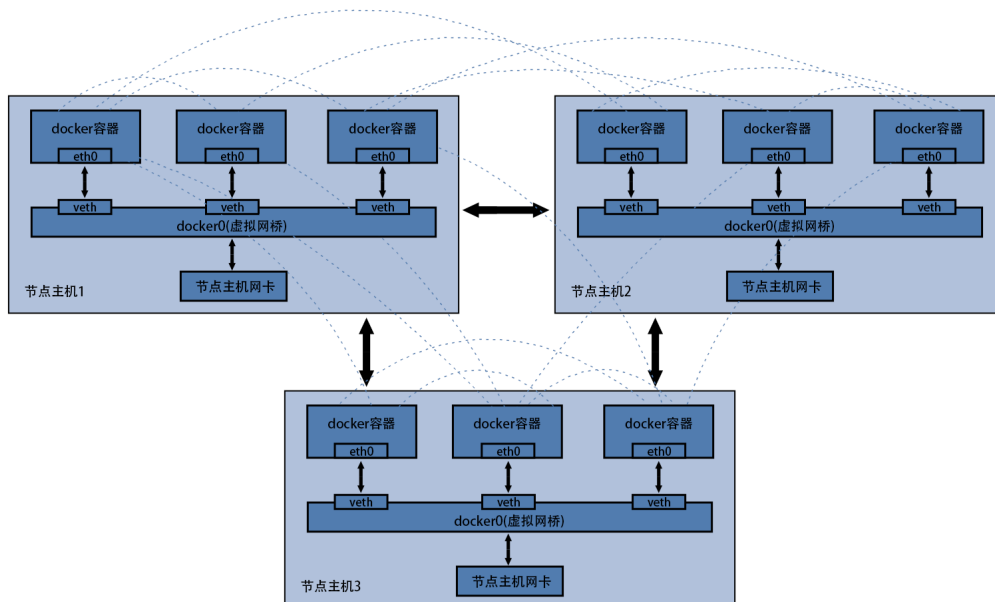
挂载硬盘后可开启 [快照备份](#) 功能。

其他设置

立即部署集群: 如果需要一次性往同一集群中添加多个节点, 可以在添加最后一个节点时再勾选此项, 以便提高添加和部署节点的效率。

容器网络

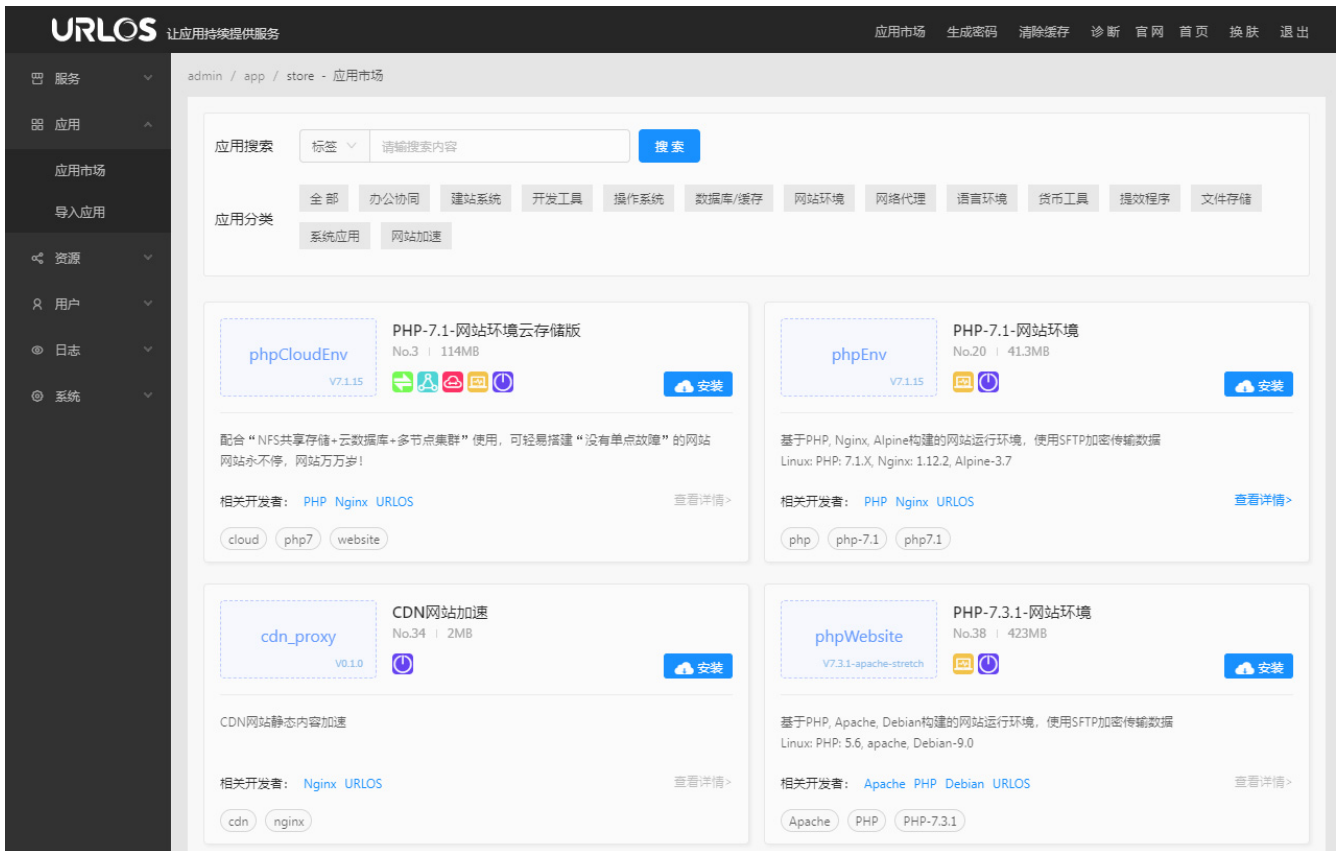
简单理解成各容器之间可相互通信的网络, 同一个容器网络下, 容器之间可相互访问。



在容器网络中, 容器之间可跨主机通信

安装应用

应用市场



应用市场提供了种类丰富、数量繁多的服务器相关应用资源，几乎可囊括Docker官方镜像市场的全部应用镜像。

安装数据库服务

以安装MySQL-5.7-数据库为例，点击 [安装](#) 按钮：



基本信息

打开 [基本信息](#) 选卡：



服务名称: 以字母开头, 由小写字母、数字和下划线组成, 设置后不可修改。为方便识别, 将服务命名为 `mysql5_7_1`, 表示MySQL5.7数据库的第1个应用。

运行节点: 请选择运行该服务的运行节点, 可选择同一集群环境下任意节点。

部署方式: 请选择部署服务的方式, **关联部署** 为强制部署本服务和相关服务, **暴力部署** 为先删除现有服务后再部署。**部署时的服务中断时间:** 智能部署<强制部署<暴力部署; **部署成功率:** 智能部署<强制部署<暴力部署。一般情况下, **推荐使用的智能部署**。

服务端口: 需要对外网开放访问限时才需要填写服务端口, 多个请用空格隔开, 如: `80 443`; 或使用“服务端口:容器端口/IP协议”的格式进行填写, 如: `8080:80/tcp 53:53/udp`, 如果是tcp协议, 则可省略“/tcp”, 即 `8080:80 53:53/udp`, mysql一般使用 `3306` 端口。

设置

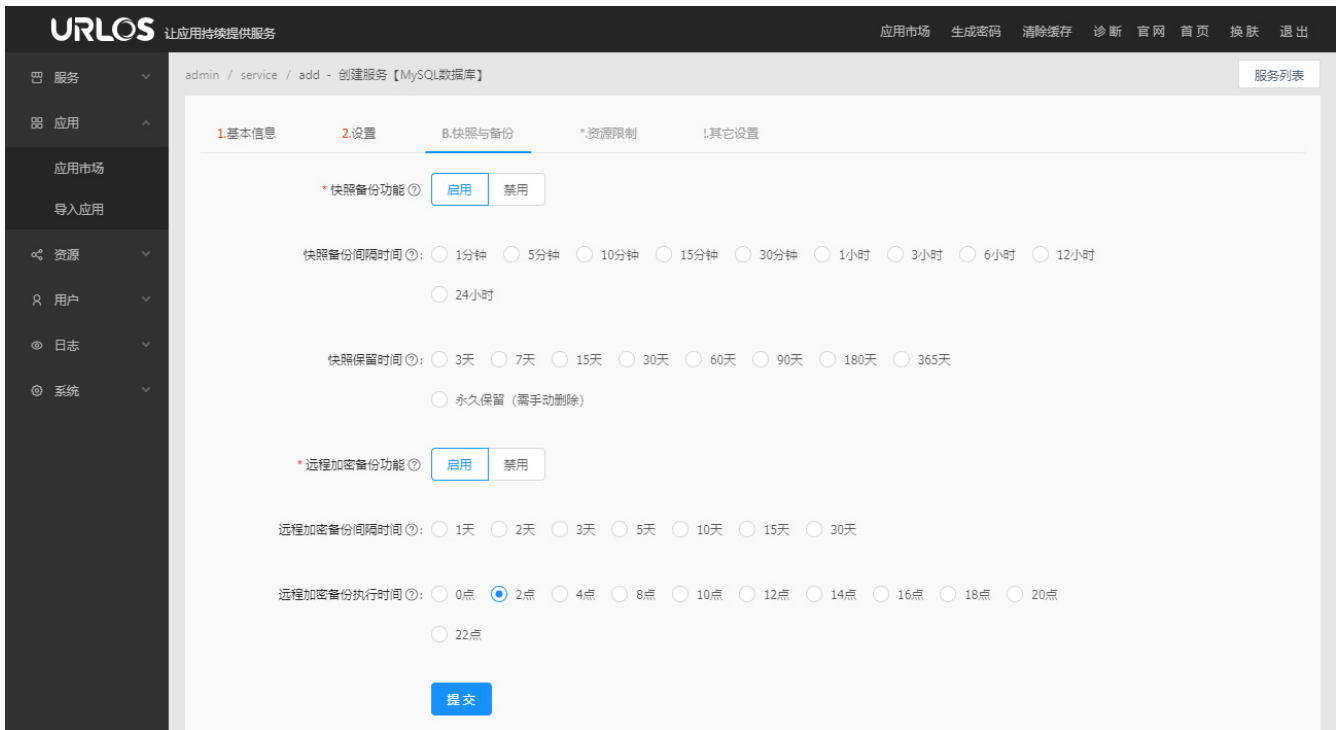
打开 **设置** 选卡:



root用户密码: 此密码为mysql应用的root用户密码。

快照与备份

打开 **快照与备份** 选卡:



在此项中可开启快照和远程加密备份功能，其中快照功能可设置快照间隔时间和快照保留时间；远程加密备份功能可设置备份间隔时间和备份执行时间。

资源限制

打开 [资源限制](#) 选卡：



该功能可限制当前应用所能使用的服务器硬件资源占比。

容器最大CPU核心数限制： 容器最大CPU核心数限制。

容器最大内存数限制： 每容器的容器最大内存数限制。注意：[过低的内存限制可能会导致某些应用程序启动失败](#)。

其他设置

打开 [其他设置](#) 选卡：



容器网络: 容器网络可理解为容器的局域网, 使用不同容器网络的服务不能互相访问, 除非应用本身支持“跨域访问”, 一般情况下, 用户无需设置。

部署优先级: 在同一个容器网络中, 当优先级小的服务未部署成功时, 其它服务将无法部署 (也即无法启动), 其值为1~99的整数。

状态: 开启或关闭

安装网站环境

以安装PHP-7.1-网站环境为例, 点击 [安装](#) 按钮:



phpWebsite
v7.1.15

PHP-7.1-网站环境

No.20 | 41.3MB

[安装](#)

基于PHP, Nginx, Alpine构建的网站运行环境, 使用SFTP加密传输数据
Linux: PHP: 7.1.X, Nginx: 1.12.2, Alpine-3.7

相关开发者: [PHP](#) [Nginx](#) [URLOS](#) [查看详情>](#)

[php](#) [php-7.1](#) [php7.1](#)

基本信息

打开 [基本信息](#) 选卡:



服务名称: 以字母开头, 由小写字母、数字和下划线组成, 设置后不可修改。为方便识别, 将服务命名为 `php7_1_1`, 表示php7.1环境的第1个应用。

运行节点: 请选择运行该服务的运行节点, 可选择同一集群环境下任意节点。

部署方式: 请选择部署服务的方式, **关联部署** 为强制部署本服务和相关服务, **暴力部署** 为先删除现有服务后再部署。**部署时的服务中断时间: 智能部署<强制部署<暴力部署; 部署成功率: 智能部署<强制部署<暴力部署。**一般情况下, **推荐使用的智能部署**。

服务端口: 需要对外网开放访问权限时才需要填写服务端口, 多个请用空格隔开, 如: `80 443`; 或使用“服务端口:容器端口/IP协议”的格式进行填写, 如: `8080:80/tcp 53:53/udp`, 如果是tcp协议, 则可省略“/tcp”, 即 `8080:80 53:53/udp`, 网站环境一般使用 `80` 端口。

网站

打开 **网站** 选卡:



网站域名: 请填写网站域名, 多个请用空格或换行符隔开, 如: `a.com www.a.com`

子目录域名: 格式为“子目录名:域名”, 多个请分行填写, 如 `bbs:bbs.abc.com` 即为bbs子目录绑定 `bbs.abc.com` 域名。

最大的并发数: 最输入Nginx的最大并发数, 一般情况下无需修改。

上传大小限制：请输入上传大小限制，单位为M，如：20M，请依照实际需求设置。

索引目录：即网站的索引目录名。如果留空，索引文件直接放在网站根目录/data/www下；如果值为public，则网站的索引目录需放在/data/www/public下。一般情况下无需填写。

数据库

打开 [数据库](#) 选卡：

The screenshot shows the '3.数据库' (Database) configuration page in the URLOS interface. The breadcrumb path is 'admin / service / add - 创建服务【PHP-7.1网站环境】'. The page has tabs for '1.基本信息', '2.网站', '3.数据库', 'B.快照与备份', 'F.上传下载', 'C.安全证书', '#.PHP配置', '%.Nginx配置', and '*.资源'. The configuration fields are:

- * 数据库主机名: 请选择数据库服务 (dropdown menu)
- * 数据库密码: (text input field)
- * 数据库字符集: gbk_chinese_ci big5_chinese_ci utf8_general_ci utf8mb4_unicode_ci
- 数据库名: 与本网站的服务名称一致 (text input field)
- 数据库用户名: 与本网站的服务名称一致 (text input field)

使用帮助:

- “数据库主机名”为“所选数据库的服务名称”，“数据库名”和“数据库用户名”为“本网站的服务名称”；
- 如有疑问，请通过SFTP下载/data/database-info.json文件查看详情！

提交

数据库主机名：请选择数据库服务。

数据库密码：请输入数据库密码。注：数据库用户名为网站的服务名称。

数据库字符集：请选择数据库字符集，一般情况下默认使用 `utf8_general_ci` 即可。

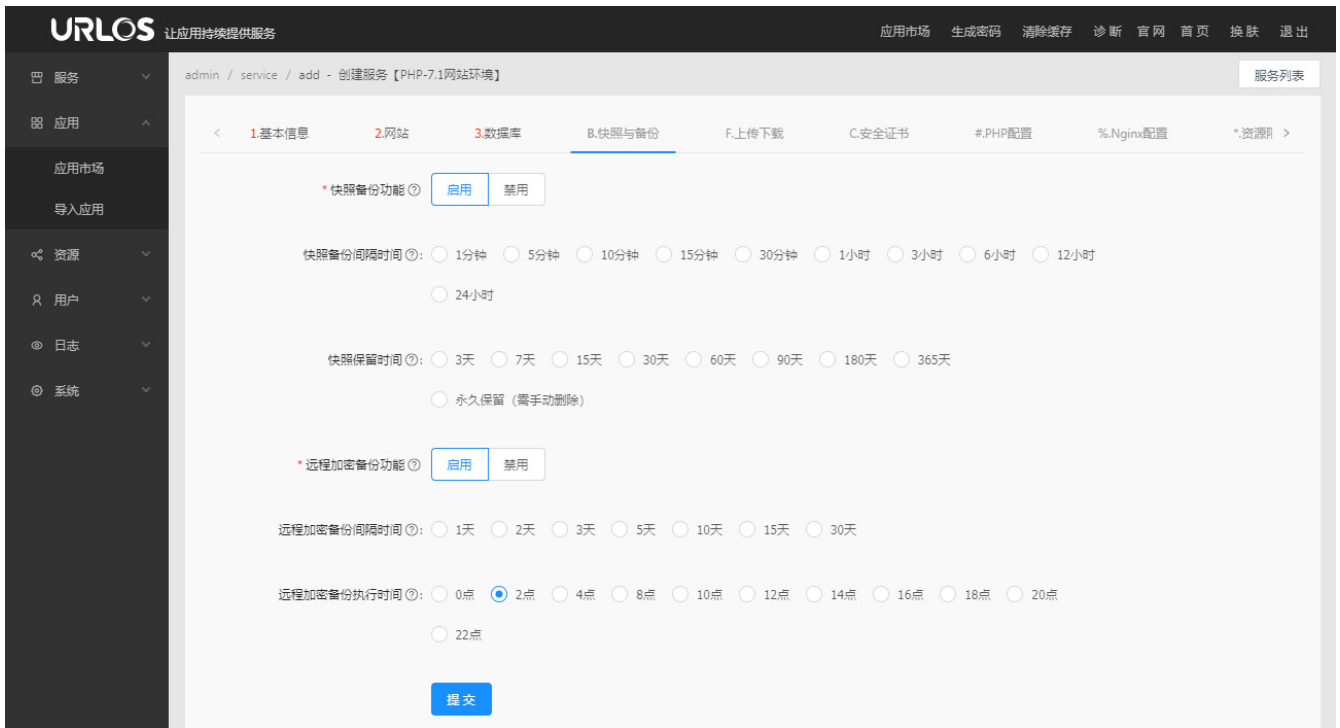
数据库名：与本网站的服务名称一致，即本例中的 `php7_1_1`。

数据库用户名：与本网站的服务名称一致，即本例中的 `php7_1_1`。

如有疑问，请通过SFTP下载 `/data/database-info.json` 文件查看详情。

快照与备份

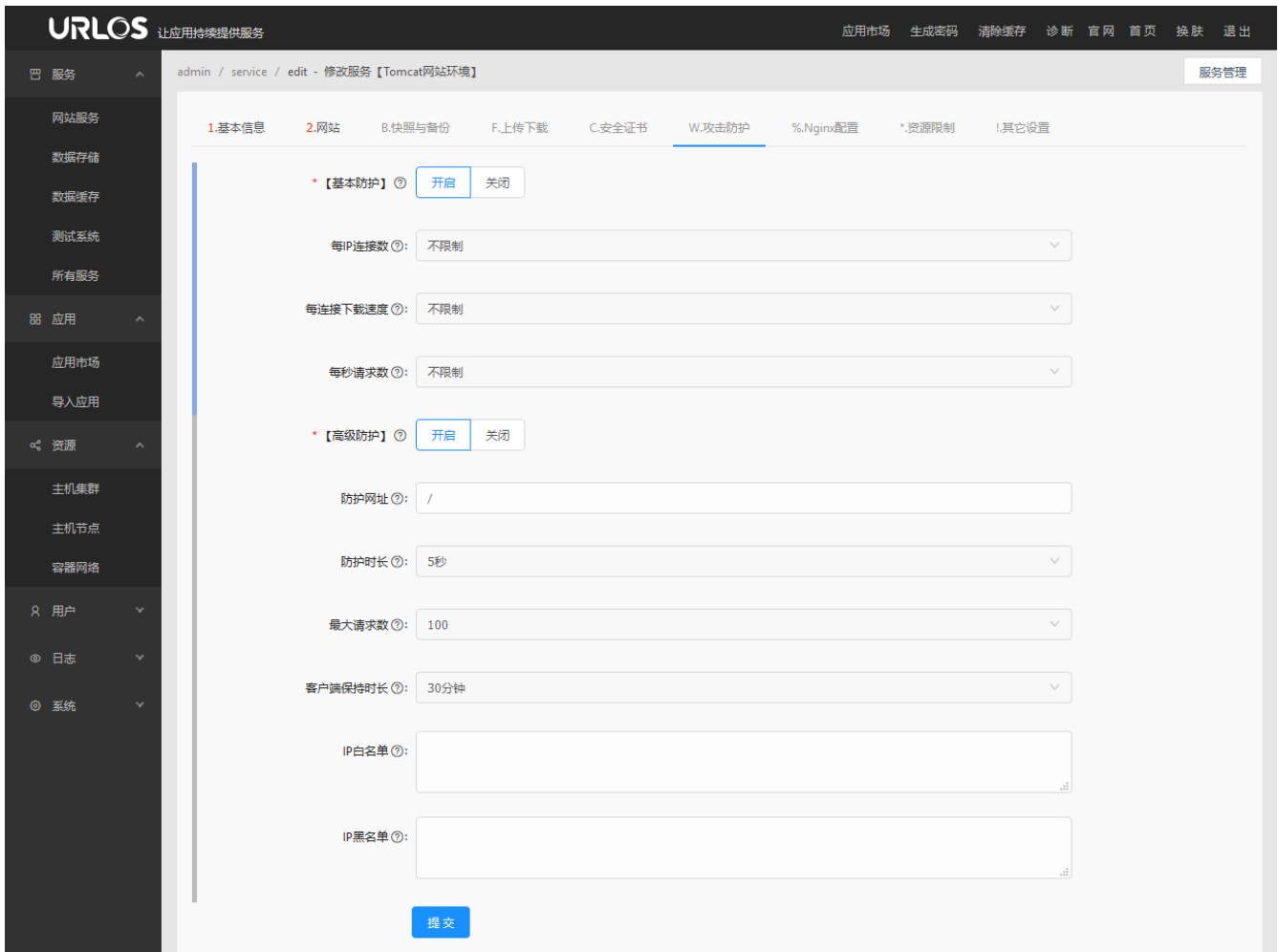
打开 [快照与备份](#) 选卡：



在此项中可开启快照和远程加密备份功能，其中快照功能可设置快照间隔时间和快照保留时间；远程加密备份功能可设置备份间隔时间和备份执行时间。

攻击防护

打开 [攻击防护](#) 选卡，初始状态默认关闭攻击防护功能，根据实际需求开启相关防护：



基本防护：基本防护功能可限制网站带宽和低级的CC攻击，如需更高级的防攻击功能，请配合高级防护使用。

每IP连接数：每个IP的最大连接数限制。

每连接下载速度：每个连接的最大下载速度限制。

每秒请求数：每IP在每秒的最大请求数。

高级防护：限制每IP在一定时间的总请求数，对非浏览器访问直接过滤，建议在遭遇恶意攻击时开启。

防护网址：支持正则匹配，默认值"/"表示防护所有网页，配置成".(php|jsp|cgi)\$"表示防护以"php|jsp|cgi"为扩展名的网址。

防护时长：请选择防护时长，一般为10秒即可。

最大请求数：一定时间内的请求数达到此数量时，即显示错误页面。

客户端保持时长：如果某客户端合法则放入白名单，保持一段时间后再重新检查。

IP白名单：白名单中的IP不需检查即可访问网站，多个IP请换行。

IP黑名单：黑名单中的IP不可访问网站，多个IP请换行。

上传下载

打开 **上传下载** 选卡，一般情况下，该功能为关闭状态，如果需要向服务器传输和管理网站文件，请点击 **开启** 按钮激活功能：



URLOS推荐使用SFTP安全文件传送协议来提高安全性。

SFTP端口：请输入SFTP端口，建议为22001，该端口不可与其他服务的端口相同。

SFTP密码：请输入SFTP密码（SFTP用户名为服务名称）。

SFTP用户名：root（不可修改）。

SFTP使用帮助：

1. 支持SSH和SFTP登录，用户名为root——此root用户与节点本身的root用户没有任何关系，可任意修改密码，请勿混淆；
2. 强烈推荐你使用URLSSH，其同时支持SSH登录和SFTP文件管理，软件下载网址：<https://www.urlos.com/urlssh>；
3. SFTP服务与网站环境不在同一个容器中，是完全隔离的，故登录后仅能管理网站数据，不能修改网站环境；
4. 网站数据在/mounts/目录下；
5. 经SFTP传输的数据是加密的，很难被监听，相比FTP更安全；

安全证书

打开 [安全证书](#) 选卡：



URLOS可设置SSL安全证书，支持手动填写SSL证书以及自动申请和自动更新SSL证书。

手动填写SSL证书：

Https证书公钥：请输入Https证书公钥（public.pem）。

Https证书私钥：请输入Https证书私钥（private.key）。

强制使用HTTPS：当存在SSL证书时，是否开启http自动跳转至https访问网站。

自动申请并更新证书：

点击 [允许](#) 即开启自动申请。

注意：

1. 无论手动和还是自动SSL，都需要在 [基本信息](#) 选卡中的 [服务端](#) 里添加 [80](#) 和 [443](#) 端口，并且在服务器中放行80和443端口；
2. [域名必须正确解析](#)，以便位于国外的LetsEncrypt官方服务器可访问并验证域名的合法性；
3. 由于受到网络等影响，自动申请证书有可能会失败，可尝试更换DNS的nameserver后也无法申请或更新证书；
4. 某些小厂的DNS解析服务可能导致申请证书失败，请尽量使用大厂的DNS解析服务！

PHP配置

打开 [PHP配置](#) 选卡：



目前PHP配置可修改最大内存、短标签、FPM最大进程数。

Nginx配置

打开 [Nginx配置](#) 选卡：



该功能允许用户对nginx配置进行适当修改，比如添加网站伪静态规则等，如果用户对Nginx配置不甚熟悉，请不要修改该配置，一旦设置出错有可能导致网站不可访问。

资源限制

打开 [资源限制](#) 选卡：



该功能可限制当前应用所能使用的服务器硬件资源占比。

单容器最大CPU核心数限制：单容器最大CPU核心数限制。

单容器最大内存数限制：每容器的单容器最大内存数限制。注意：[过低的内存限制可能会导致某些应用程序启动失败](#)。

其他设置

打开 [其他设置](#) 选卡：



容器网络：容器网络可理解为容器的局域网，使用不同容器网络的服务不能互相访问，除非应用本身支持“跨域访问”，一般情况下，用户无需设置。

部署优先级：在同一个容器网络中，当优先级小的服务未部署成功时，其它服务将无法部署（也即无法启动），其值为1~99的整数。

状态：开启或关闭

安装CDN网站加速

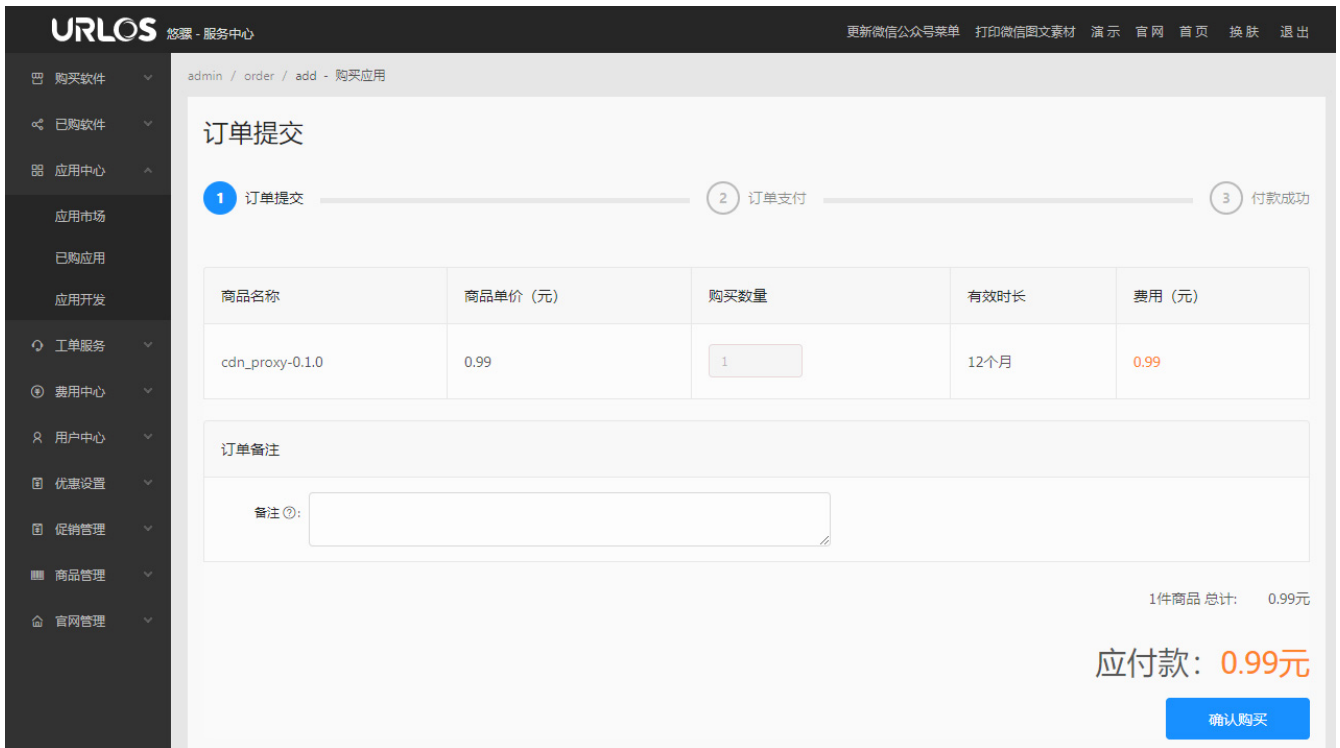
URLOS付费版支持CDN网站加速，免费版本暂不开放支持。

购买应用

URLOS付费版用户如需使用CDN网站加速功能，请登录URLOS服务中心(<https://www.urlos.com>)，在 [应用中心](#) > [应用市场](#) 中找到**CDN网站加速**应用：



点击 [安装](#) 进入应用购买页面：



购买成功即可安装CDN网站加速。

基本信息

打开 [基本信息](#) 选卡：



服务名称：以字母开头，由小写字母、数字和下划线组成，设置后不可修改。为方便识别，将服务命名为 `php7_1_1`，表示php7.1环境的第1个应用。

运行节点：请选择运行该服务的运行节点，可选择同一集群环境下任意节点。

部署方式：请选择部署服务的方式，[关联部署](#) 为强制部署本服务和相关服务，[暴力部署](#) 为先删除现有服务后再部署。部署时的服务中断时间：[智能部署](#)<[强制部署](#)<[暴力部署](#)；部署成功率：[智能部署](#)<[强制部署](#)<[暴力部署](#)。一般情况下，[推荐使用的智能部署](#)。

服务端口：需要对外网开放访问权限时才需要填写服务端口，多个请用空格隔开，如：`80 443`；或使用“服务端口:容器端口/IP协议”的格式进行填写，如：`8080:80/tcp 53:53/udp`，如果是tcp协议，则可省略“/tcp”，即 `8080:80 53:53/udp`，网站环境一般使用 `80` 端口。

网站

打开 [网站](#) 选卡：



网站域名：请填写网站域名，多个请用空格或换行符隔开，如：`a.com www.a.com`

源站设置

打开 [源站设置](#) 选卡：



源站网址：请输入源站网址，如：<https://www.urlos.com>或<http://www.urlos.com>

源站IP：请输入源站IP，如：192.168.44.33

替换主机头：是否替换主机头（提示：某些源网站可能不支持替换主机头）

目录缓存

打开 [目录缓存](#) 选卡：



可设置源站中哪些目录需要缓存，并设置缓存更新时间，目前最多可设置5个缓存目录。推荐将静态文件目录设置为缓存目录，如：js文件目录、css文件目录、图片文件目录、静态HTML目录等。缓存成功后，访客通过CDN节点域名访问网站时，可从当前节点中读取静态文件，从而起到加速目的。

文件缓存

打开 [文件缓存](#) 选卡：



可指定需要缓存的文件类型以及缓存更新时间。

格式为：需要缓存的文件后缀名，留空则使用默认值，多个用“|”隔开，如：jpg|png|css|js|mov|mp4|mp3

安全证书

打开 [安全证书](#) 选卡：



CDN也可设置SSL安全证书，支持手动填写SSL证书以及自动申请和自动更新SSL证书。

手动填写SSL证书：

Https证书公钥：请输入Https证书公钥（public.pem）。

Https证书私钥：请输入Https证书私钥（private.key）。

强制使用HTTPS：当存在SSL证书时，是否开启http自动跳转至https访问网站。

自动申请并更新证书：

点击 [允许](#) 即开启自动申请。

注意：

1. 无论手动和还是自动SSL，都需要在 [基本信息](#) 选卡中的 [服务端口](#) 里添加 **80** 和 **443** 端口，并且在服务器中放行80和443端口；
2. [域名必须正确解析](#)，以便位于国外的LetsEncrypt官方服务器可访问并验证域名的合法性；
3. 由于受到网络等影响，自动申请证书有可能会失败，可尝试更换DNS的nameserver后也无法申请或更新证书；
4. 某些小厂的DNS解析服务可能导致申请证书失败，请尽量使用大厂的DNS解析服务！

Nginx配置

打开 [Nginx配置](#) 选卡：



该功能允许用户对nginx配置进行适当修改，比如添加网站伪静态规则等，如果用户对Nginx配置不甚熟悉，请不要修改该配置，一旦设置出错有可能导致网站不可访问。

CDN调优

打开 [CDN调优](#) 选卡：

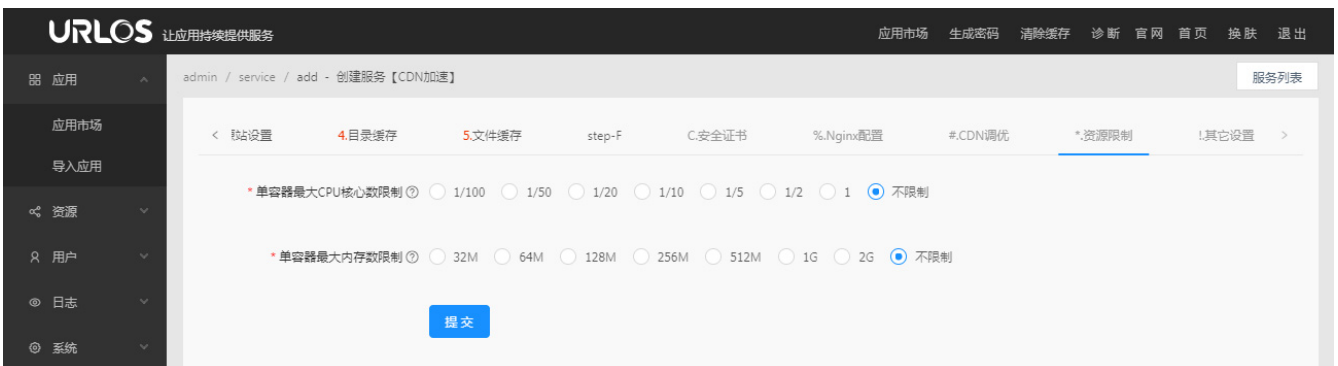


缓存目录层级：可设置缓存目录深度。

缓存文件总大小：设置缓存文件大小时，请考虑好磁盘的容量，否则可能会导致磁盘不够用。

资源限制

打开 [资源限制](#) 选卡：



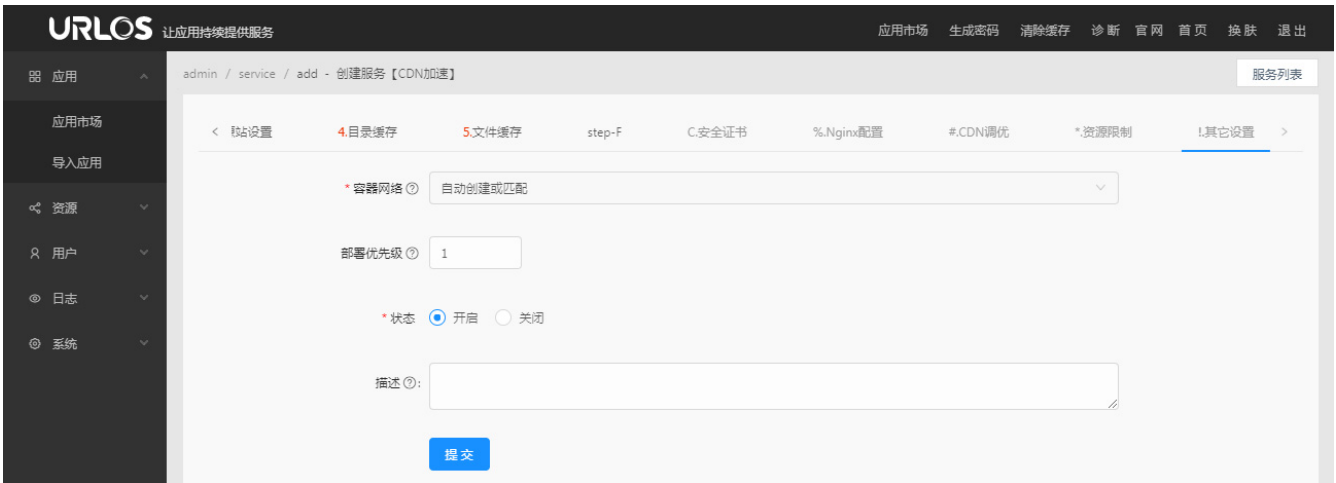
该功能可限制当前应用所能使用的服务器硬件资源占比。

单容器最大CPU核心数限制：单容器最大CPU核心数限制。

单容器最大内存数限制：每容器的单容器最大内存数限制。注意：[过低的内存限制可能会导致某些应用程序启动失败](#)。

其他设置

打开 [其他设置](#) 选卡：



容器网络：容器网络可理解为容器的局域网，使用不同容器网络的服务不能互相访问，除非应用本身支持“跨域访问”，一般情况下，用户无需设置。

部署优先级：在同一个容器网络中，当优先级小的服务未部署成功时，其它服务将无法部署（也即无法启动），其值为1~99的整数。

状态：开启或关闭

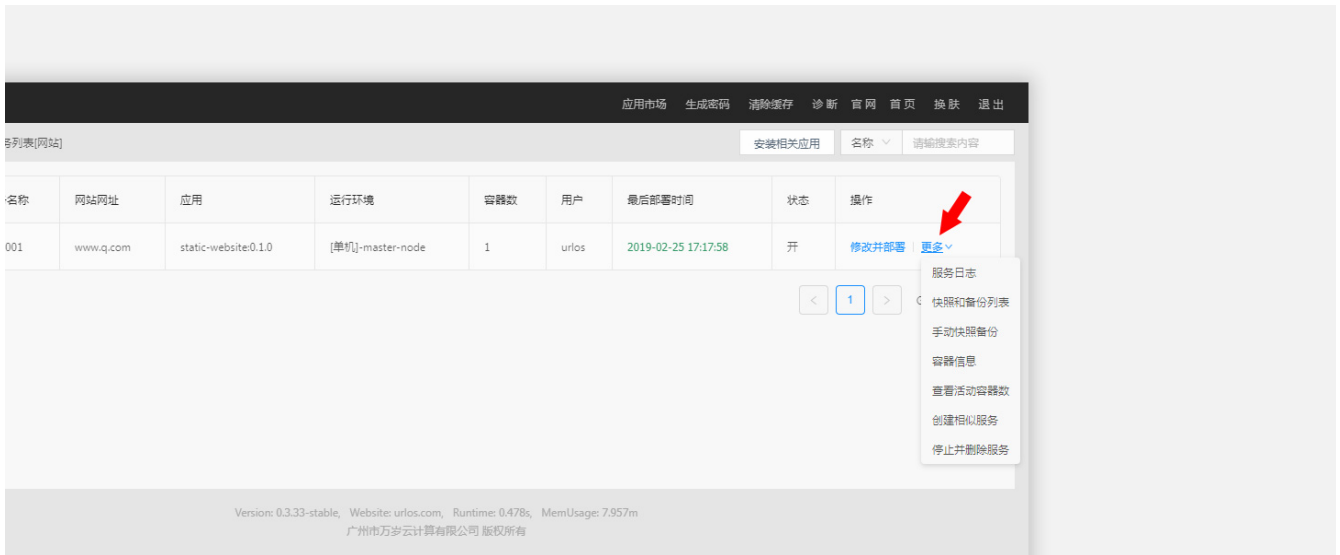
服务管理

网站服务

网站服务列表专门为网站类型的服务而设置，其中可包括网站环境类型（php网站环境、java网站环境、静态网站环境等）、网站系统类型（CMS系统、博客系统、BBS系统、电商系统等），可在此集中管理。

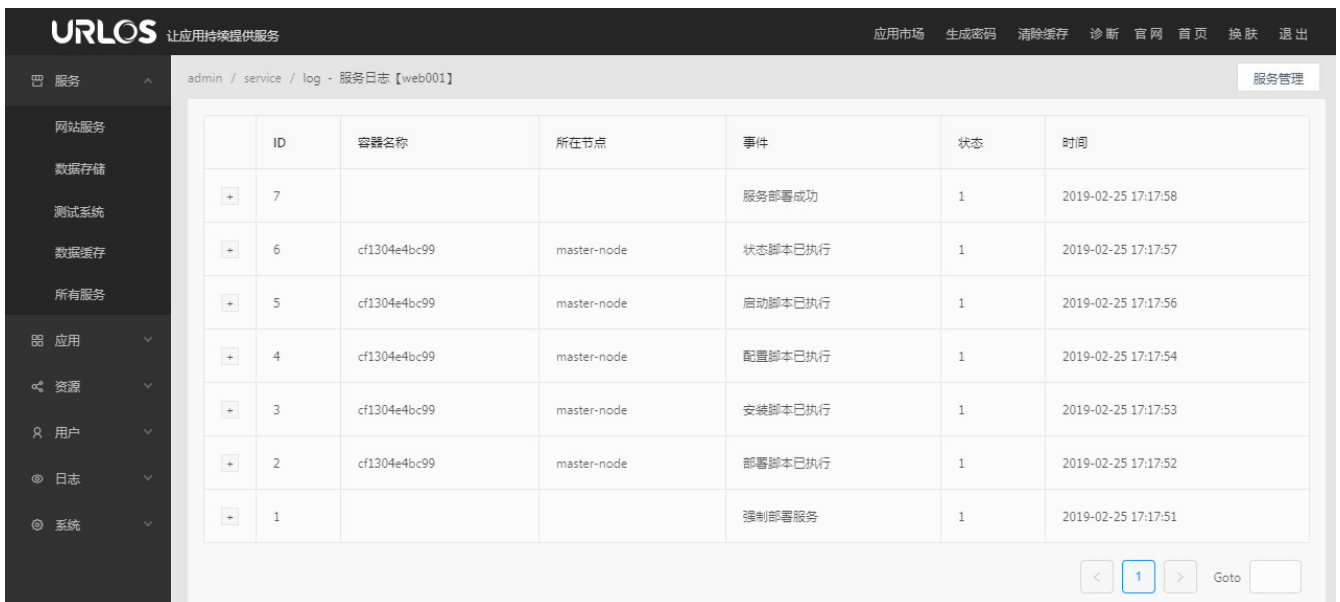


这里重点说明如何管理已创建的服务，点击 [更多](#) 显示菜单，包括服务日志、快照和备份列表、手动快照备份、容器信息、查看活动容器数、创建相似服务、停止并删除服务。



服务日志

打开 [服务日志](#)：



在服务日志中，可以查看当前服务产生的所有事件、对应容器名称、对应节点以及时间节点，通过分析服务日志，可以帮助用户检查和监控服务健康状况。

快照和备份列表

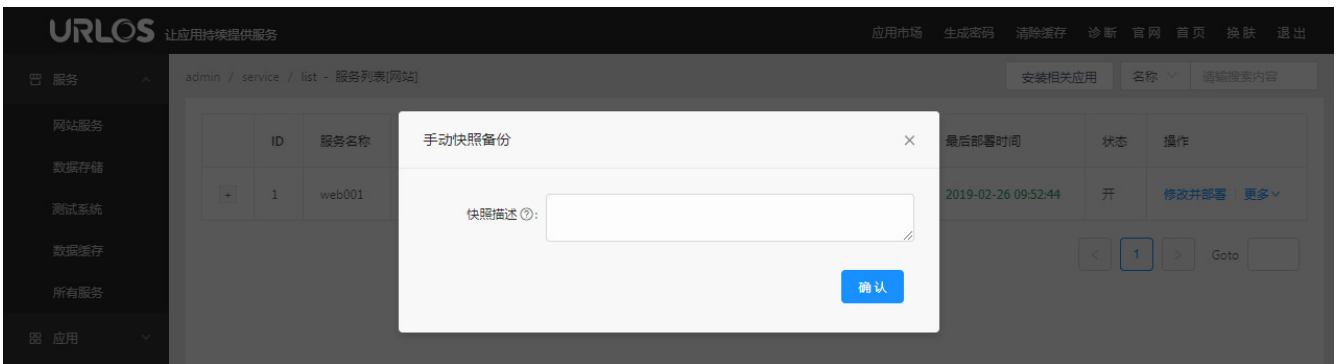
打开 [快照和备份列表](#)：



该功能允许用户管理快照与远程加密备份，可对快照备份执行 [恢复](#) 与 [删除](#) 操作。

手动快照备份

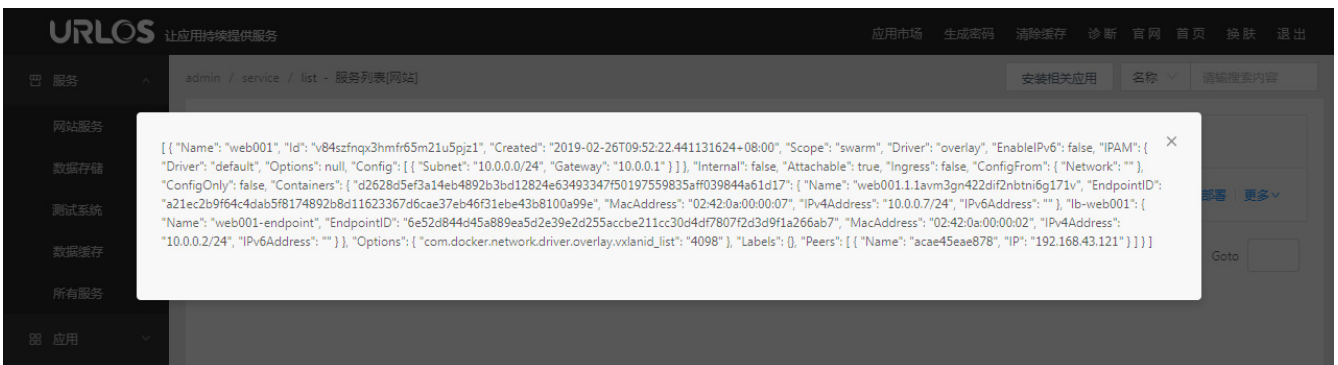
用户可随时执行手动快照备份，点击 [手动快照备份](#)，弹出对话框：



点击 [确认](#) 按钮即可完成快照备份。

容器信息

打开 [容器信息](#)：



该功能可展示当前容器的相关信息，阅读容器信息需要用户具备一定的docker容器基础知识，普通用户可不必要理解。

查看活动容器数

查看当前服务的活动容器数量。



创建相似服务

可直接创建一个相似的服务实例，被创建的服务是区别于当前服务的，是一个全新的服务。



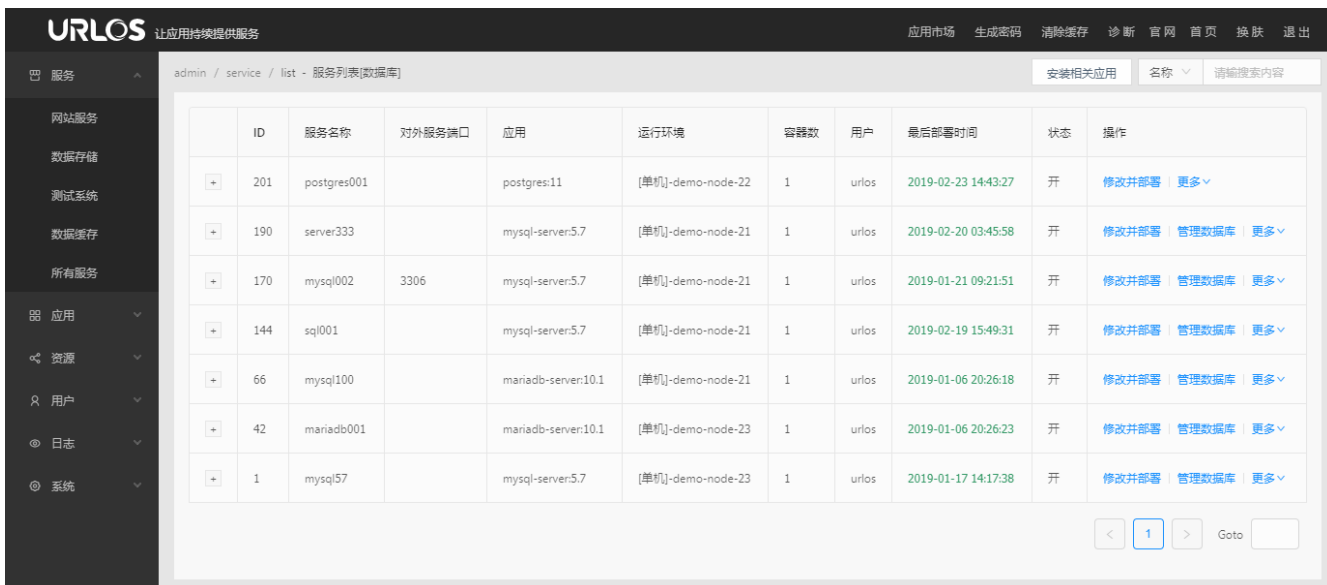
停止并删除服务

停止当前服务，并且删除服务。



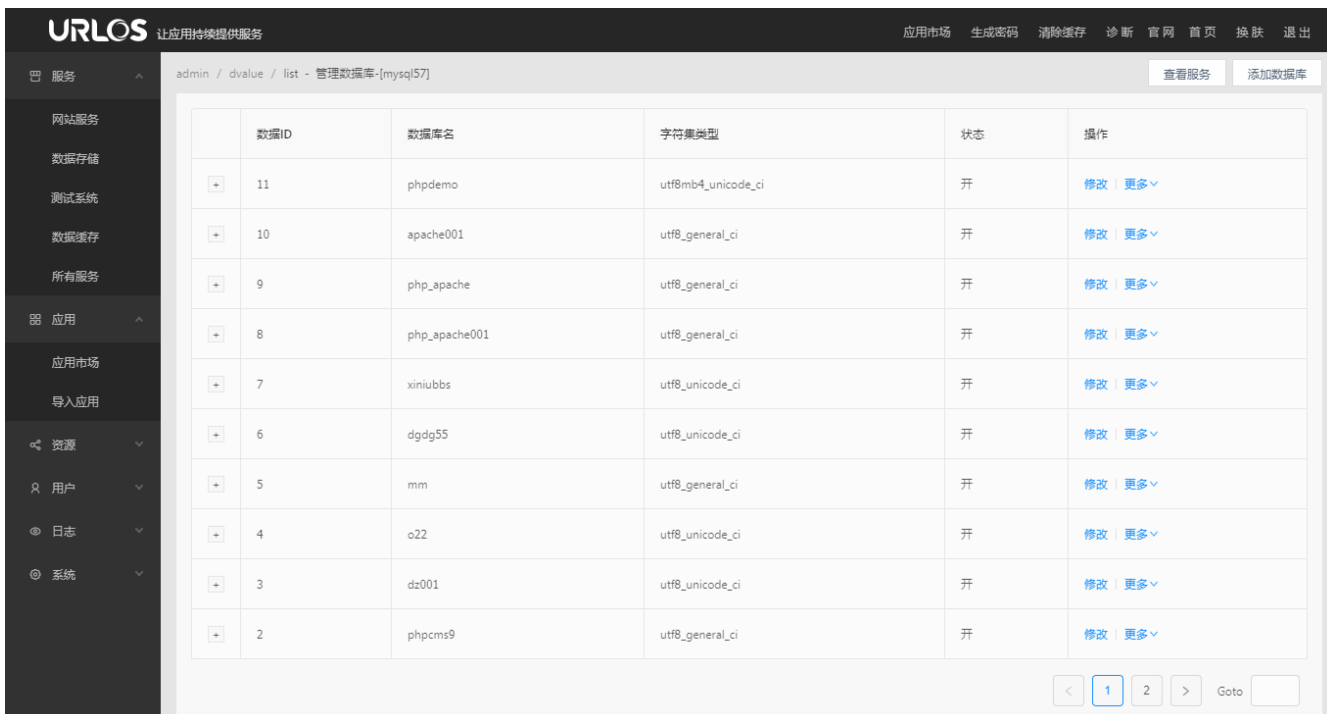
数据存储

数据存储列表专门为数据库类型的服务而设置，可在此集中管理。



管理数据库

选择一个数据库服务，点击打开 [管理数据库](#) 功能：



该功能提供了对数据库的添加、查看、修改、删除等功能。

添加数据库

点击界面右上方的 [添加数据库](#) 按钮，进入添加数据库功能：



通过该功能，用户可手动添加数据库。

修改数据库

点击 [修改](#) 按钮：



通过该功能，可修改数据库密码、字符集类型、描述、状态。



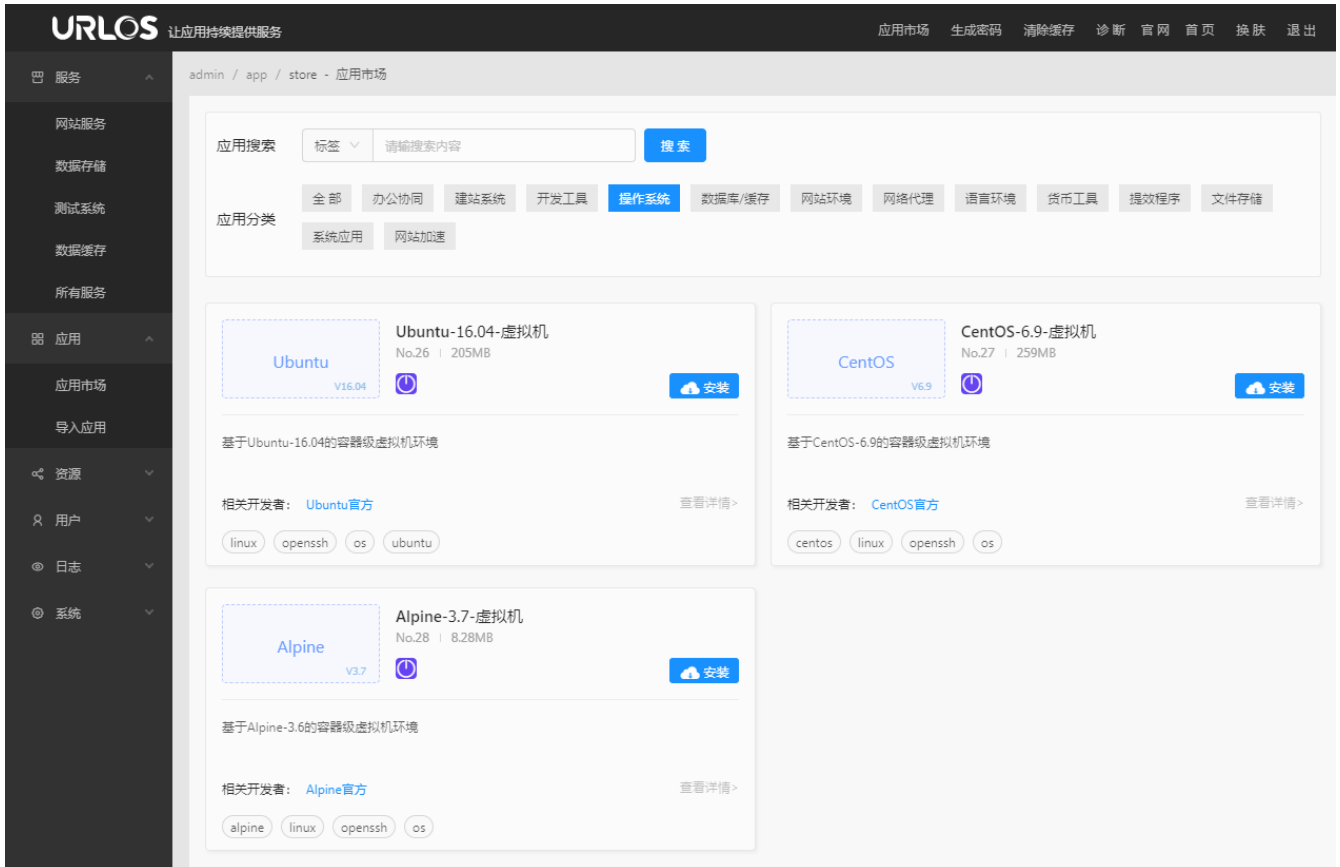
删除

可删除数据库。

测试系统

测试系统列表是专为各种Linux轻量系统(如Ubuntu、Alpine、CentOS等)的管理而设置，测试系统仅用于测试使用，[不可作为生产环境](#)使用！

开发人员、测试人员可在应用市场中快速安装各种系统对程序代码进行开发调试、测试，不会破坏宿主机环境，非常方便。



数据缓存

对数据缓存类型的应用进行集中管理，如Memcached。

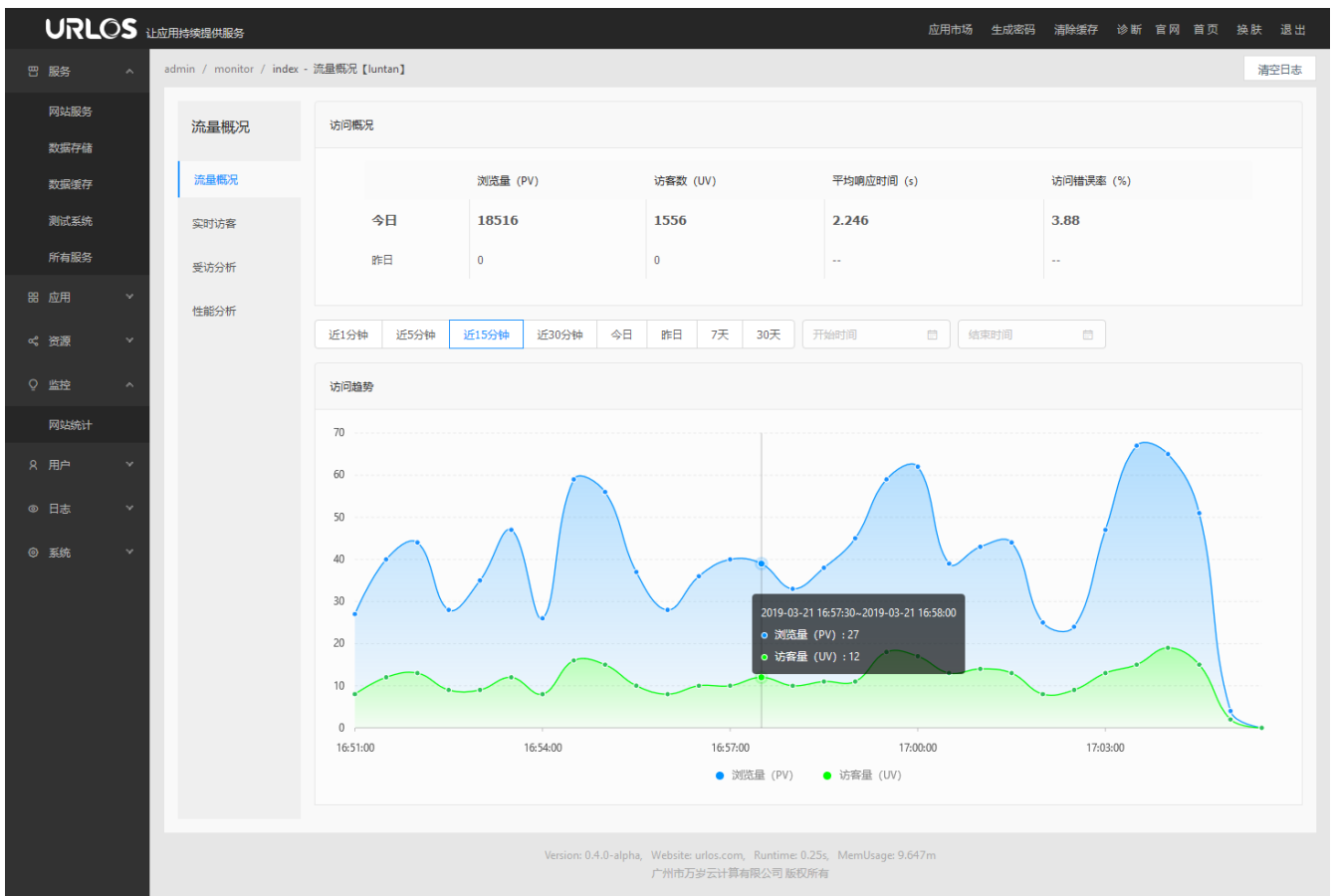
所有服务

该列表展示了所有的服务，除了之前以做归类的网站服务、数据存储、测试系统、数据缓存，还有未归类的代理服务、SFTP服务等。

监控分析

网站统计

可查看网站服务的访问情况，功能包括流量概况、实时访客、受访分析、性能分析：



开启监控分析功能

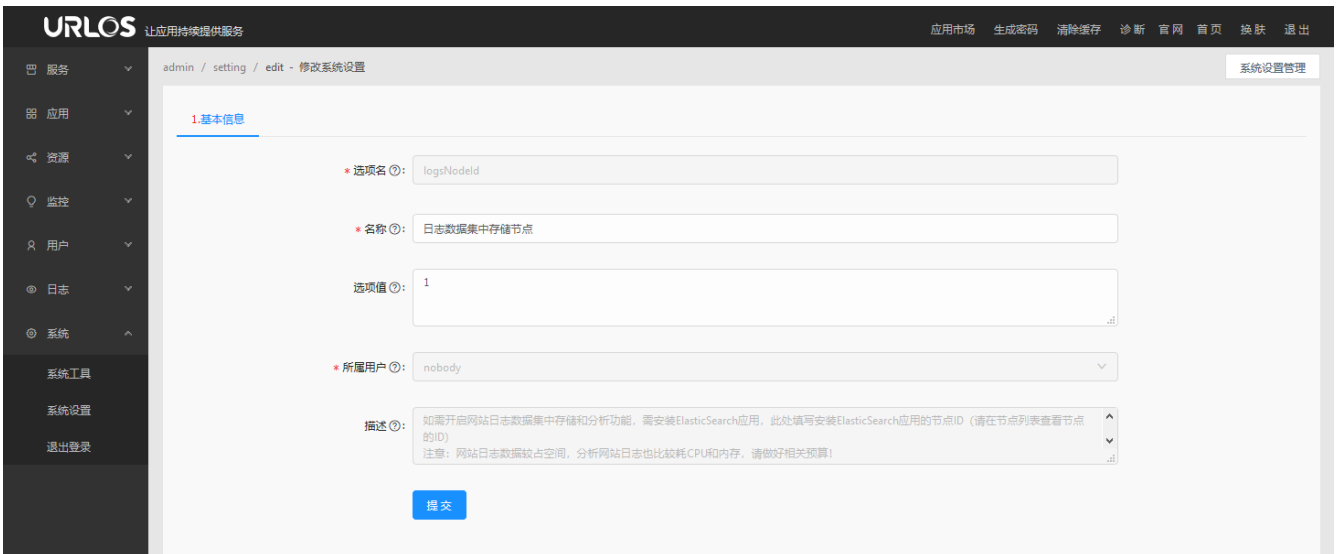
在服务器内存达到2G或者更高的情况下，才可开启监控分析功能，URLOS免费版暂不开发此功能。

在左侧系统菜单中找到 [系统设置](#) 选项，在系统设置列表中点击修改 [日志数据集中存储节点](#)：

The screenshot shows the 'admin / setting / list - 系统设置列表' page in URLOS. It features a table with system configuration items. The table has columns for '名称' (Name), '选项' (Option), '选项值' (Option Value), '所属用户' (User), '修改时间' (Modification Time), and '操作' (Action). The '日志数据集中存储节点' (Log data central storage node) is highlighted in blue.

名称	选项	选项值	所属用户	修改时间	操作
程序异常重启次数	restartMaxRetries	0	nobody	2019-03-20 23:03:14	修改
绑定域名提示	showBindDomainHelp	1	nobody	2019-02-21 13:44:44	修改
日志数据集中存储节点	logsNodeId	1	nobody	2019-03-21 15:34:54	修改
页面动画类型	animationType	none	nobody	2018-09-05 18:35:19	修改
强密码验证开关	forceStrongPasswords	1	nobody	2019-02-25 14:29:23	修改
容器退出时的等待时间	scriptsErrorExitWaitTime	60	nobody	2019-01-04 22:01:04	修改
Docker镜像加速源	dockerImageMirror		nobody	2019-02-02 10:59:44	修改
Linux软件镜像源	linuxSoftwareMirrors		nobody	2018-10-27 12:17:27	修改
通知检查间隔时间	notificationCheckTime	2	nobody	2017-12-16 21:43:59	修改

进入修改界面：



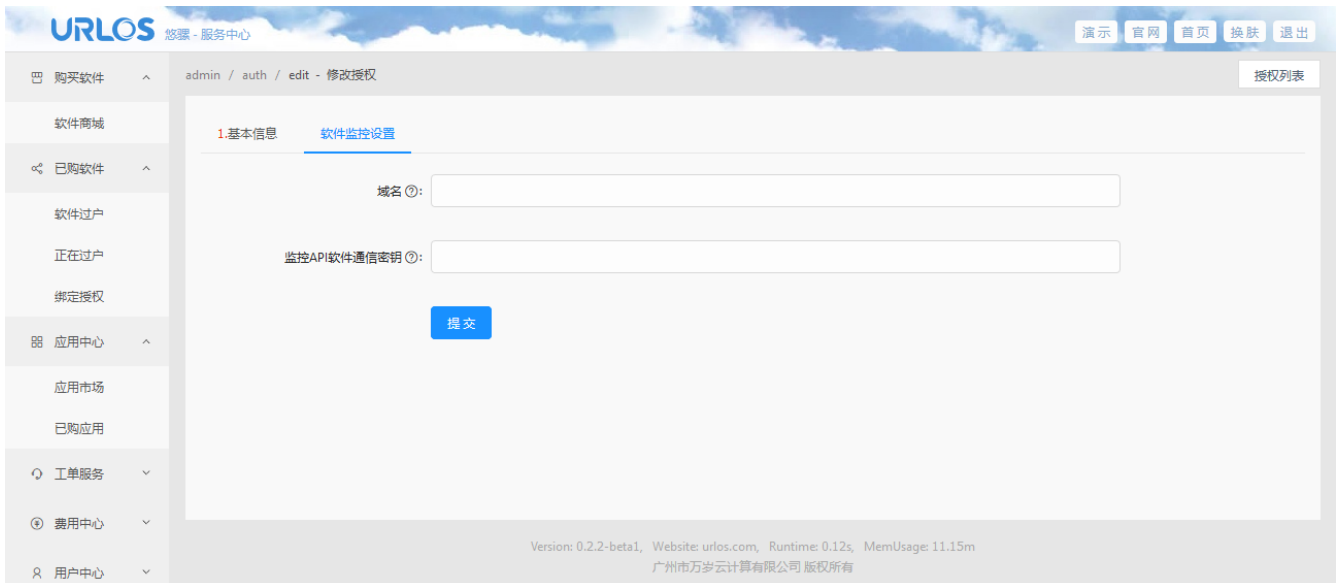
找到 **选项值** 一项，在此设置日志数据保存在哪个节点主机中，将节点的ID填入即可，点击提交即开启监控分析功能。如何查看节点ID：在左侧菜单打开 **资源 > 主机节点**，在节点列表查看节点ID。

微信查看主机监控

登录URLOS服务中心，找到 **已购软件>绑定授权** 菜单，打开授权码绑定列表，点击 **修改** 操作：

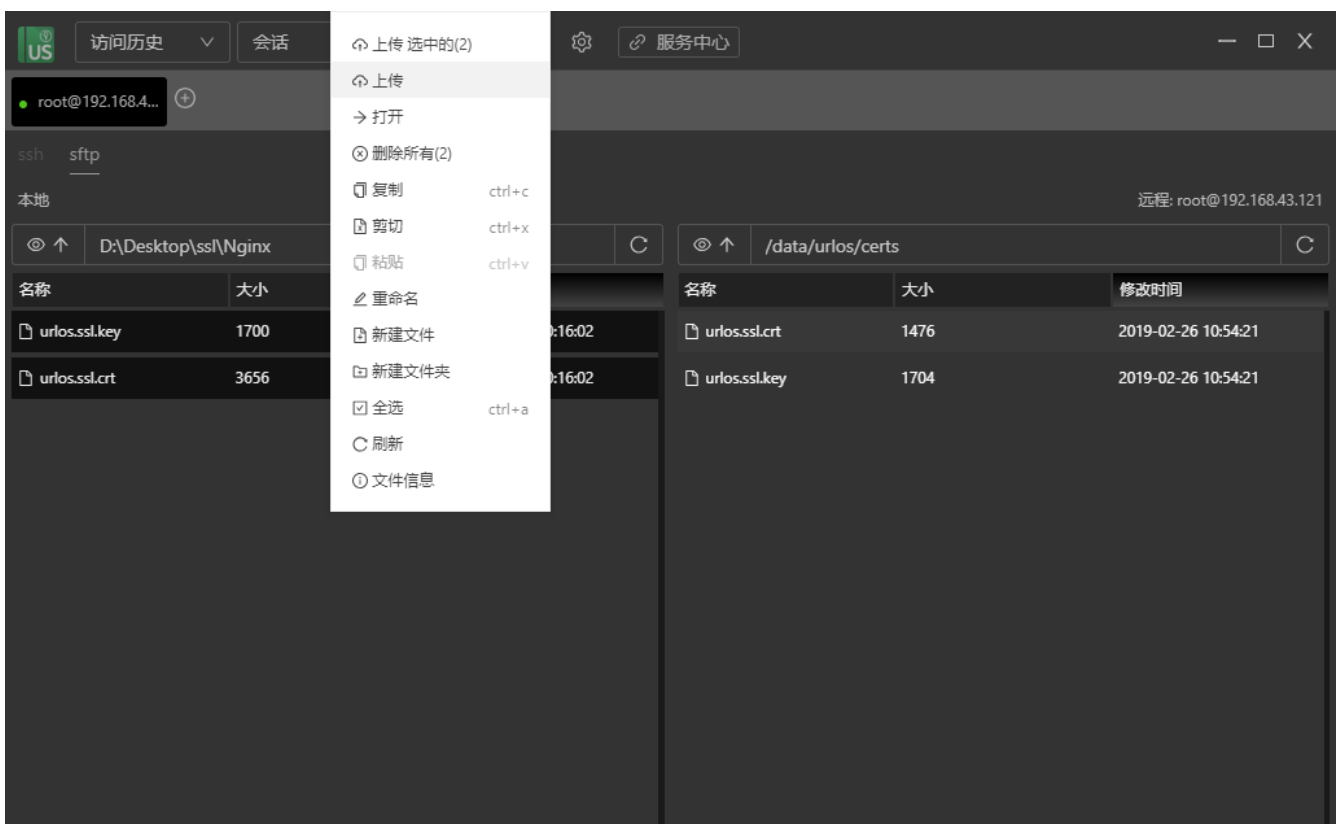


进入修改界面后，打开 **软件监控设置** 选卡，填写域名与监控API软件通信密钥。这里所填写的域名必须拥有合法有效的SSL证书，且将SSL证书文件替换URLOS默认的SSL证书。



证书替换方法：

- 1、使用SFTP工具连接主机，打开URLOS的证书目录“`/data/urllos/certs`”；
- 2、将你的SSL证书文件名修改为“`urllos.ssl.key`”和“`urllos.ssl.crt`”；
- 3、把改名后的证书文件上传替换URLOS的默认证书。

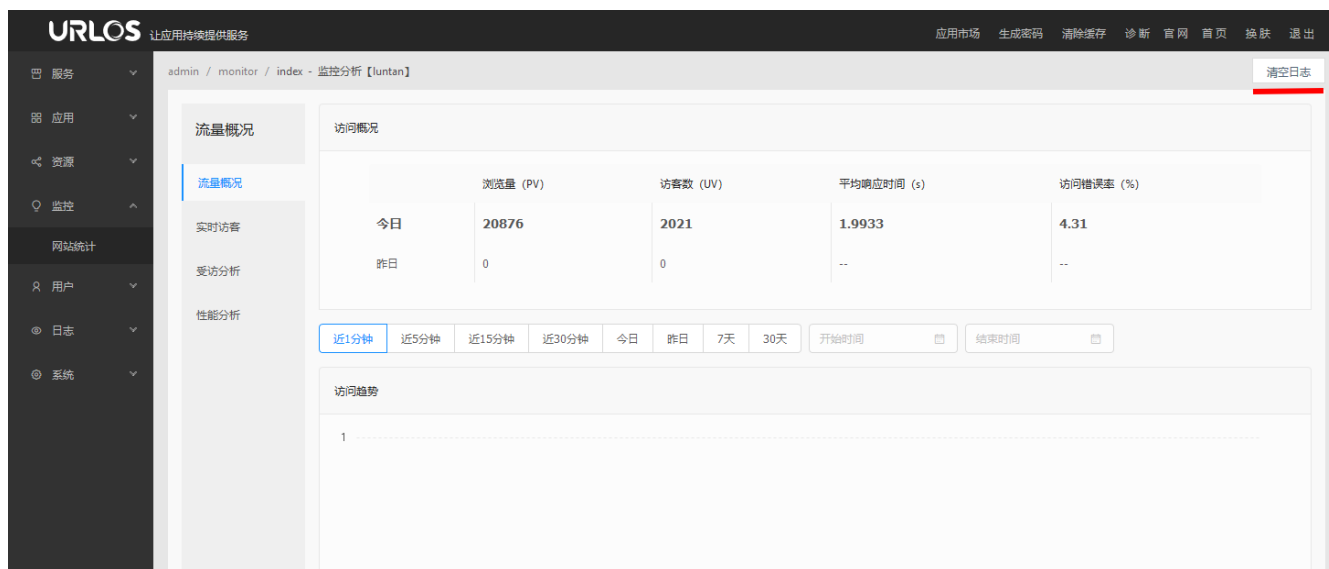


注意：主机请开放9966端口，否则提交时会提示域名验证超时！

完成以上配置后，打开【URLOS】公众号，点击公众号底部菜单“[主机管理](#) > [主机监控](#)”即可查看相关监控信息。

网站日志清理

由于监控功能将会记录集群中所有主机的网站访问日志，并将所有日志集中存储在某一主机节点中，若网站流量巨大，则日志文件也会非常庞大，因此URLOS提供了日志清理功能，可让用户选择清理自定义时段内的日志数据。



选择需要清理的时间段：



日志管理

任务计划

可通过任务计划列表查看各项任务的执行情况，帮助排查相关故障。

URLOS 让应用持续提供服务

admin / cron / list - 任务计划列表

ID	日志	执行结果	通知状态	UriQuery	用户	开始执行时间	耗时(秒)
5334	[services_stfp_phpcom9]正在准备部署服务...	cancel	已取消任务	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=205&um=D7tpe	nobody		
5333	[servicesmysq57]正在准备部署服务...	cancel	已取消任务	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=1&um=D7tpe	nobody		
5332	[swarmIP100.X]正在部署并检查集群状态...	cancel	已取消任务	cswarm&as=swarmDeployCron&v=0.3.34&id=1&um=4zewor	nobody		
5331	[services_provy_3]正在准备部署服务...	cancel	已取消任务	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=4&um=D7tpe	nobody		
5330	[services_provy_2]正在准备部署服务...	cancel	已取消任务	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=3&um=D7tpe	nobody		
5329	[services_provy_1]创建服务时发生了错误!	error	已执行&已显示	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=2&um=D7tpe	nobody	2019-02-26 16:04:31	23
5328	所有副本均执行正常, 整体已部署100%	success	已执行&已显示	cserviceCheck&as=checkDeployScriptExecResultCron&v=0.3.34&id=8&serviceType=service&um=1F3qIM&retry=2	nobody	2019-02-26 16:04:02	26
5327	已更新服务, 开始部署服务...[force]	info	已执行&已自动关闭	cserviceDeploy&as=serviceDeployCron&v=0.3.34&id=8&um=D7tpe	urlos	2019-02-26 16:03:46	13
5326	[swarmIP100.X]集群已部署, 状态良好	success	已执行&已自动关闭	cswarm&as=swarmDeployCron&v=0.3.34&id=1&um=4zewor	urlos	2019-02-26 16:03:17	26
5325	正在更新代理服务器配置并重新代理服务...	info	已执行&已显示	cserviceCheck&as=checkDeployScriptExecResultCron&v=0.3.34&id=3&serviceType=container&um=1F3qIM&cronid=5323	nobody		

1 2 3 4 5 ... 534 Goto

系统日志

系统日志记录了URLOS程序系统运行的相关事件，帮助用户了解URLOS运行情况。

URLOS 让应用持续提供服务

admin / log / list - 日志列表

ID	日志	UriQuery	用户	IP	时间
5642	同步最新应用[43]	csron&as=exec&um=2PzNTh	nobody		2019-02-27 10:01:00
5641	缓存已清空	cs-tool&as=flushCache&um=1uVjy	urlos	27.18.34.252	2019-02-27 10:00:41
5640	登录成功	csuser&as=loginCheck&um=3zXBg	urlos	27.18.34.252	2019-02-27 10:00:12
5639	登录成功	csuser&as=loginCheck&um=3zXBg	urlos	27.38.0.134	2019-02-27 09:19:54
5638	登录成功	csuser&as=loginCheck&v=0.3.34&um=3zXBg	urlos	113.68.254.32	2019-02-27 08:52:49
5637	未设置日志数据库存储节点, 请在系统设置的logsNodeId选项中设置节点的ID.	cserviceVisit&as=view&serviceid=189&um=2zUtg	urlos	113.111.82.248	2019-02-27 07:20:00
5636	登录成功	csuser&as=loginCheck&um=3zXBg	urlos	113.111.82.248	2019-02-27 07:19:29
5635	同步最新应用[43]	csron&as=exec&um=2PzNTh	nobody		2019-02-27 00:36:43
5634	登录成功	csuser&as=loginCheck&um=3zXBg	urlos	115.212.114.93	2019-02-26 23:46:53
5633	登录成功	csuser&as=loginCheck&um=3zXBg	urlos	183.95.180.224	2019-02-26 21:05:21

1 2 3 4 5 ... 565 Goto

###快速入门###

5分钟从0开始创建网站

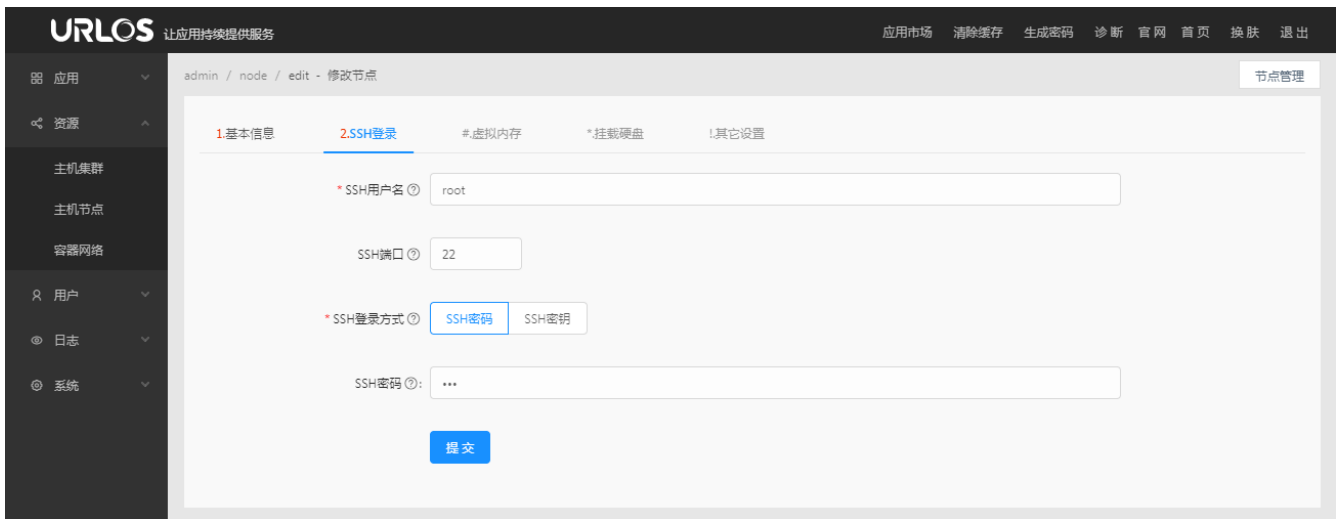
URLOS安装完成后，如何从0开始创建一个网站？以下将演示以最快捷的方式创建网站，以创建wordpress博客网站为例：

1.添加集群和节点

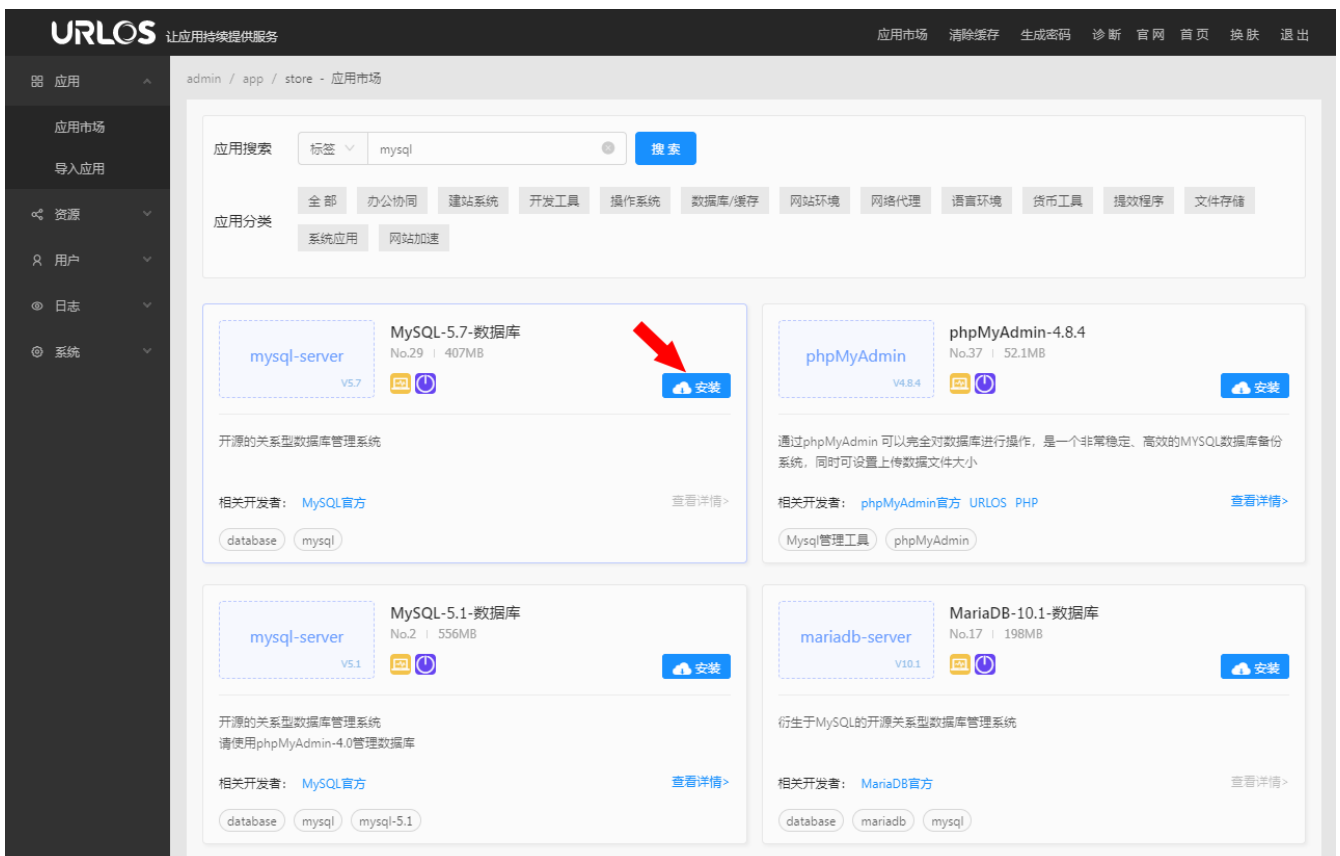
如果当前仅有一个服务器(节点)，可使用URLOS提供的 [自动添加集群和节点](#) 功能。



此时，只需请填写SSH密码或SSH密钥即可：



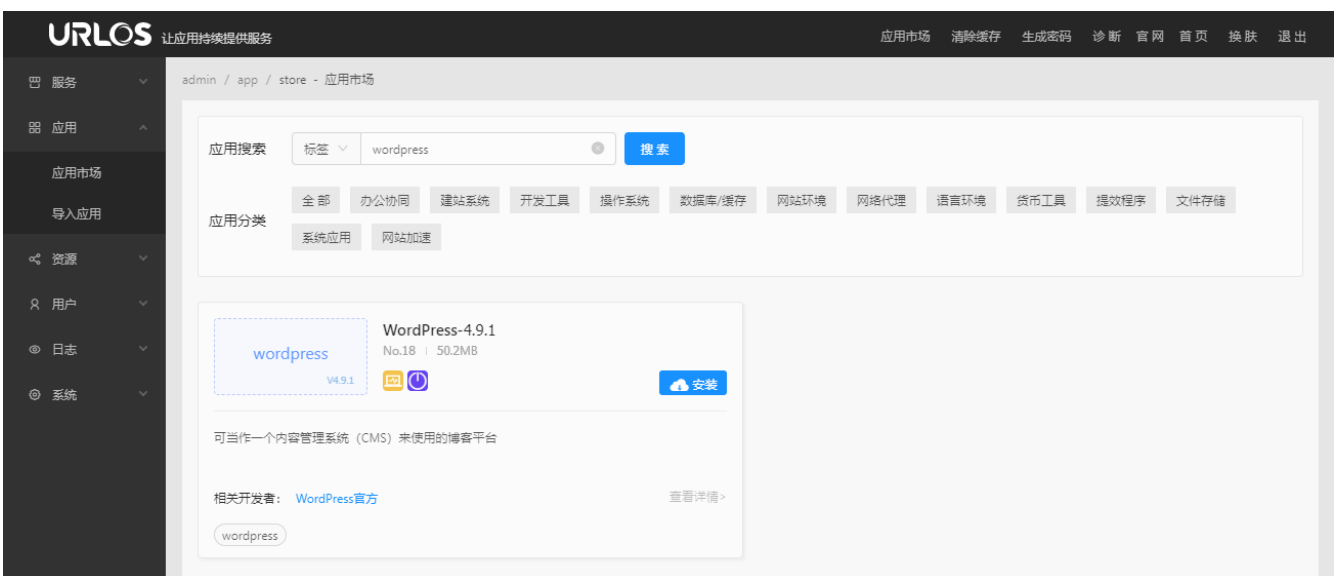
集群和节点添加完成之后，打开应用市场，先安装MySQL数据库服务：



打开创建MySQL数据库界面，在 **基本信息** 中填写服务名称、选择运行节点、部署方式默认即可、服务端口填写为3306即可，然后在 **设置** 中输入数据库密码，提交。



完成了数据库服务的创建，在应用市场中找到wordpress，点击安装：



在wordpress创建页面中完善 **基本信息**、**网站**、**数据库** 相关信息，点击提交：



待部署完成后，在浏览器中输入域名，博客网站访问成功，此刻，使用URLOS在最短的时间里成功创建了一个博客网站。

外部连接MySQL数据库

以上篇创建的MySQL和wordpress为例，打开网站服务，找到myblog这个网站，点击 **修改并部署**：



在数据库选卡中，可以找到 [数据库主机名](#)、[数据库名](#)、[数据库用户名](#) 等相关信息：



外部连接时，把 [数据库主机名](#) 更改为 [IP地址](#) 即可。

通过命令行进入容器和退出容器

举例，如何进入wordpress应用的容器内部。

使用SSH客户端连接服务器：

```
Connecting to 192.168.43.121:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+J'.

Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-87-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

211 个可升级软件包。
122 个安全更新。

New release '18.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Feb 27 15:06:46 2019 from 192.168.43.140
root@master-node:~#
```

输入以下命令，查看正在运行的容器：

docker ps

```
root@master-node:~# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
5efd33ff171    urlos/nginx:1.14.2-alpine-curl     "/urlos/keeper"        20 minutes ago Up 20 minutes 0.0.0.0:80->80/tcp
117247cb8801   urlos/wordpress-cn:4.9.1          "/urlos/keeper"        20 minutes ago Up 20 minutes
598c3f09490    urlos/mysql-server:5.7            "/urlos/keeper"        31 minutes ago Up 31 minutes
f48e388bea87   urlos/urlos:latest                 "sh /entrypoint.sh"    About an hour ago Up About an hour 0.0.0.0:9001->9001/tcp, 0.0.0.0:9966->9966/tcp, 0.0.0.0:9968->9968/tcp
root@master-node:~#
```

在列表中可以看到wordpress对应的容器ID是：`117247cb8001`，容器名称是：`myblog.1.i4dnv8wrq6lpuusu5yfkrmwh5`，注意：每次重启容器，容器ID会改变，容器名称则不会改变。

如何进去这个容器的内部？可使用“`docker exec -it 容器名称 /bin/sh`”命令进入容器内部：

```
docker exec -it myblog.1.i4dnv8wrq6lpuusu5yfkrmwh5 /bin/sh
```

进入容器后，输入`ls`命令，此刻发现这个容器内部的目录结构与宿主机非常相似：

```
root@master-node:~# docker exec -it myblog.1.i4dnv8wrq6lpuusu5yfkrmwh5 /bin/sh
/ # ls
bin      etc      media    proc     sbin     tmp      var
data     home    mnt      root     srv      urlos
dev      lib     mounts  run      sys      usr
/ #
```

继续尝试进入目录：`/mounts/myblog/data/www/`，发现博客网站文件，这些文件就存在于容器中：

```
/ # ls
bin      etc      media    proc     sbin     tmp      var
data     home    mnt      root     srv      urlos
dev      lib     mounts  run      sys      usr
/ # cd /mounts/myblog/data/www/
/mounts/myblog/data/www # ls
index.php
license.txt
phpinfo-3edbf2fd0de89efb33a6a947690c9d3b.php
readme.html
wp-activate.php
wp-admin
wp-blog-header.php
wp-comments-post.php
wp-config-sample.php
wp-config.php
wp-content
wp-cron.php
wp-includes
wp-links-opml.php
wp-load.php
wp-login.php
wp-mail.php
wp-settings.php
wp-signup.php
wp-trackback.php
xmlrpc.php
/mounts/myblog/data/www #
```

值得注意的是：任何对容器内部的修改，只在当前有效，如重启容器后，所有的修改都会恢复成原来的状态。

输入`exit`命令即可退出容器：

```
/mounts/myblog/data/www # exit
root@master-node:~#
```

制作自己的Docker镜像

1. 下载ubuntu-16.04的docker镜像:

```
docker pull ubuntu:16.04
```

下载完成后使用命令 `docker images` 查看镜像

2. 运行镜像

```
docker run -itd --name test_ubuntu -p 80:80 -v /data/ww:/var/www/html ubuntu:16.04
```

注: `--name` 自定义容器名, 不使用此参数的话, 自动生成容器名; `--p` 端口映射, 母机端口: 容器端口; `--v` 挂载目录 母机目录: 容器目录;

3. 进入容器

```
docker exec -it test_ubuntu bash
```

4. 执行命令 `apt-get update`

5. 安装nginx

```
apt-get install -y nginx
```

nginx目录默认安装在 `/etc/nginx`, 可以使用 `ls /etc/nginx` 查看

6. 执行命令 `exit` 退出容器

7. 将容器打包成镜像

```
docker commit test_ubuntu xiaoming/nginx:ubuntu-16.04
```

test_ubuntu 这个可以是容器名、容器ID xiaoming/nginx:ubuntu-16.04 镜像仓库: 镜像标签 可以执行命令 `docker commit --help` 查看参数

8. 登录docker

```
docker login
```

之后输入账户密码 (需要到 hub.docker.com 注册账户)

9. 将镜像上传到docker云端

```
docker push xiaoming/nginx:ubuntu-16.04
```

xiaoming 这个必须是注册账户的ID,在注册账户时填写 如果需要修改镜像名, 可使用

```
docker tag xiaoming/nginx:ubuntu-16.04 xiaoqiang/nginx:ubuntu-16.04
```

使用此命令 `docker tag --help` 可以查看具体参数

Linux常用的解压命令

在Linux中最常见的就是zip和tar这两种压缩文件，下面介绍这两种压缩文件如何解压：

1、zip文件如何解压？

可使用“`unzip 文件名`”命令，如：

```
root@ubuntu:~# unzip test.zip
Archive: test.zip
  extracting: test.txt
root@ubuntu:~# ls
test.txt test.zip
root@ubuntu:~# █
```

2、tar文件如何解压？

可使用“`tar -xf 文件名`”命令，如：

```
root@ubuntu:~# tar -xf test.tar
root@ubuntu:~# ls
test.tar test.txt
root@ubuntu:~# █
```