# NextCloud 部署及配置文档

## 环境准备

NextCloud 通过 Docker 即可完成部署，部署前确保满足以下要求：

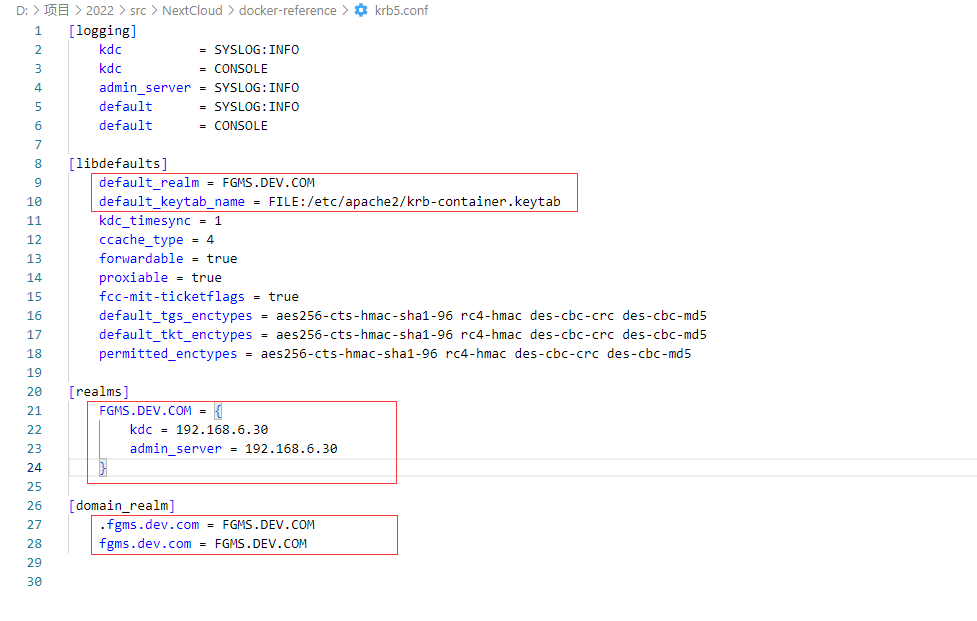
* 需要宿主机加入 AD 域控或者可以访问 AD 域服务器
* 需要为 NextCloud 申请对应的域名和证书，证书包含一个 .crt 文件和一个 .key 文件
* 需要在 .env 文件里维护环境变量NEXTCLOUD\_SERVER\_NAME，取值为上一步申请的域名
* NextCloud 使用 MySQL 数据库存储数据，如需自定义，请修改以 MYSQL\_ 开头的环境变量
* 需要知道域服务的名称，如：fgms.dev.com，以及域服务器的IP，如：192.168.6.30
* 需要通过域服务器生成一个 keytab 文件，请参考“SSO 集成”这部分内容

## 环境部署

首先，修改 .env 文件，按照实际的环境信息进行配置：



接下来，修改 /docker-reference/ 目录下的 krb5.conf 文件：

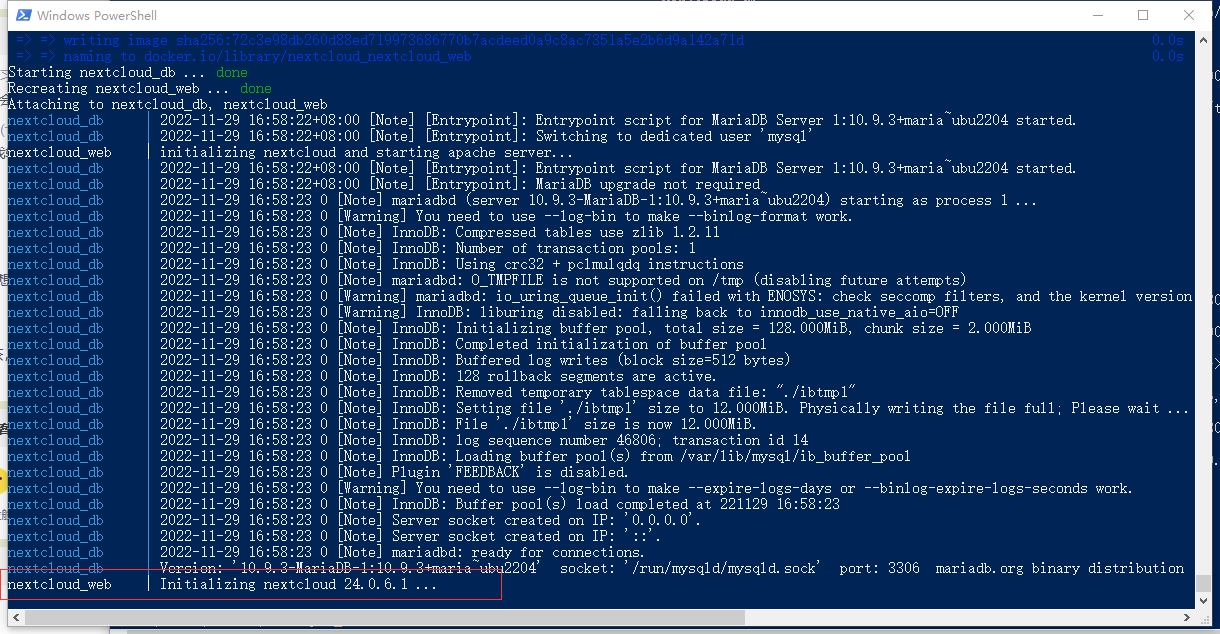


假设当前的域服务器名称为fgms.dev.com，则维护default\_realm为 FGMS.DEV.COM。请注意，该值必须全部大写。这里，可以参考“SSO 集成”这部分生成一个 keytab 文件，放置在 /docker-reference/ 目录下即可，default\_keytab\_name 字段请保持默认值不变。

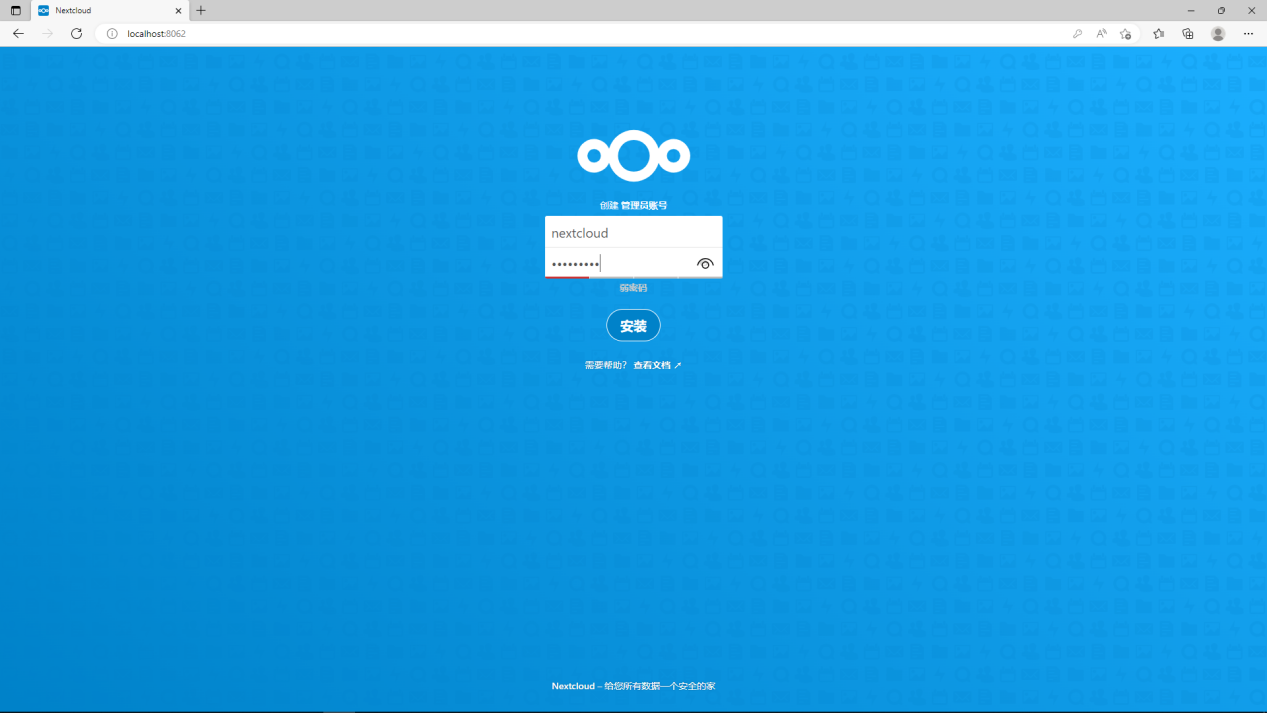
接下来，配置一个 realms 节点，节点的值应该与上一步中维护的 default\_realm 一致，对应地，下方的kdc 和 admin\_server 字段请配置为域服务器的 IP 地址。

最后，参照示例，将小写的域名形式映射到大写的域名形式，完成 Kerberos 客户端的配置，这一步配置错误会导致容器无法启动，因为容器启动时需要通过 Kerberos 初始化一次票据。

在完成以后步骤后，直接通过 docker-compose up 命令即可，容器启动后会自动完成初始化，首次初始化大约需要 5 分钟时间。部署前请检查 .env 文件的环境变量是否与当前环境一致，



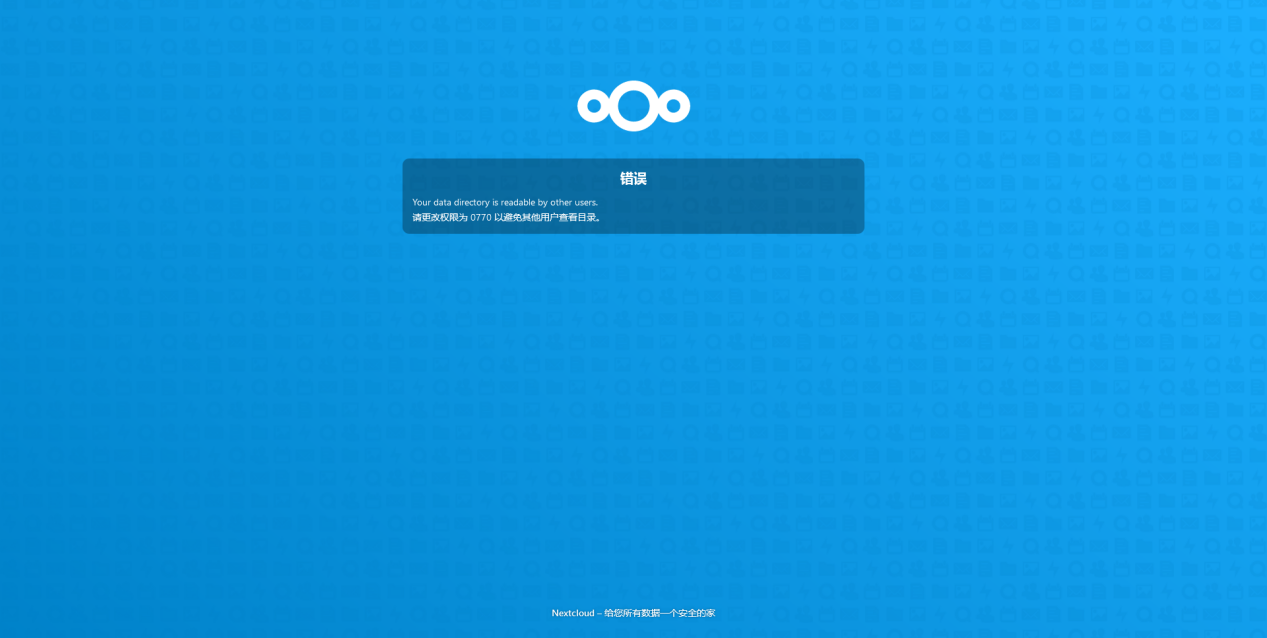
默认情况下，可以使用 [https://localhost:4433](https://<NEXTCLOUD_SERVER_NAME>:4433) 或者 [http://localhost:8061](http://<NEXTCLOUD_SERVER_NAME>:8061) 访问 NextCloud。此时，只需要按照界面提示，一直点击下一步，直到创建管理账号这一步，管理员账号对后续步骤非常重要，建议使用nextcloud/nextcloud 这组信息进行创建和登陆。



点击“安装”按界面提示，等待程序自动完成安装。



此时，如果出现下面的画面，请修改 NextCloud 的配置文件：/config/config.php。默认情况下，该文件会在宿主机上挂载出来，默认的挂载目录为：/data/nextcloud/。

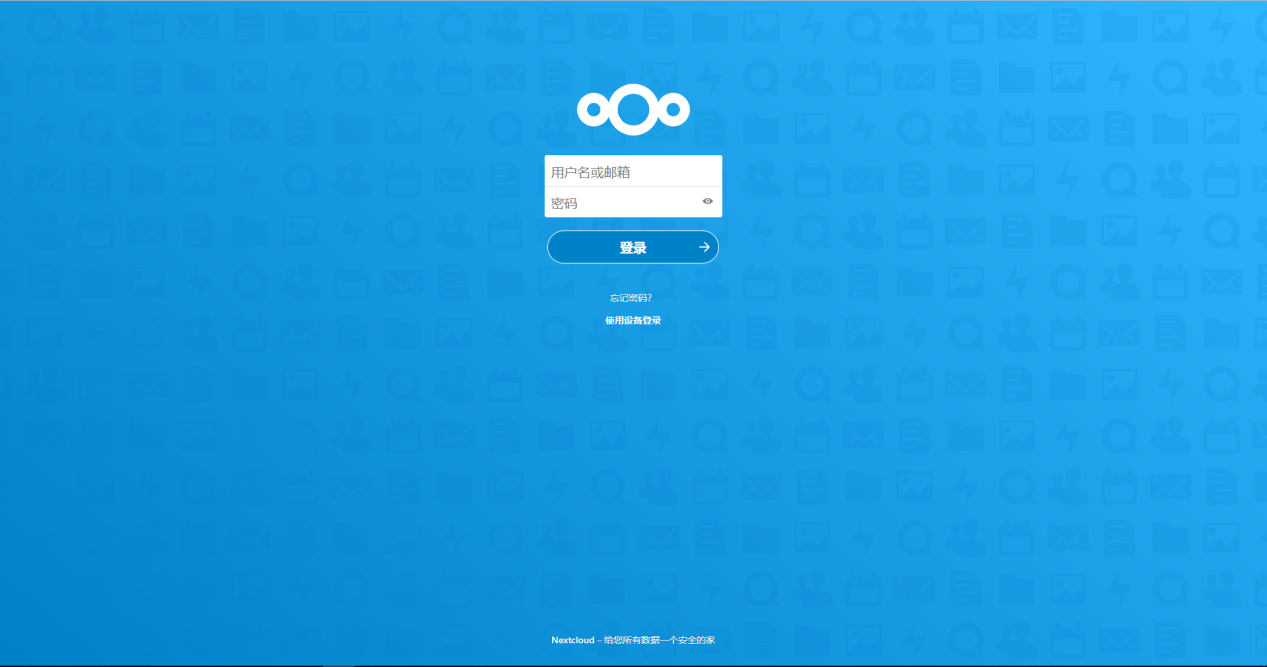


打开 /config/config.php 文件，修改配置项check\_data\_directory\_permissions为 false，如无此配置项，请自行添加此配置项，如下图所示：



如希望使用域名访问 NextCloud，请在trusted\_domains 节点加增加对应的域名，如下图所示： 

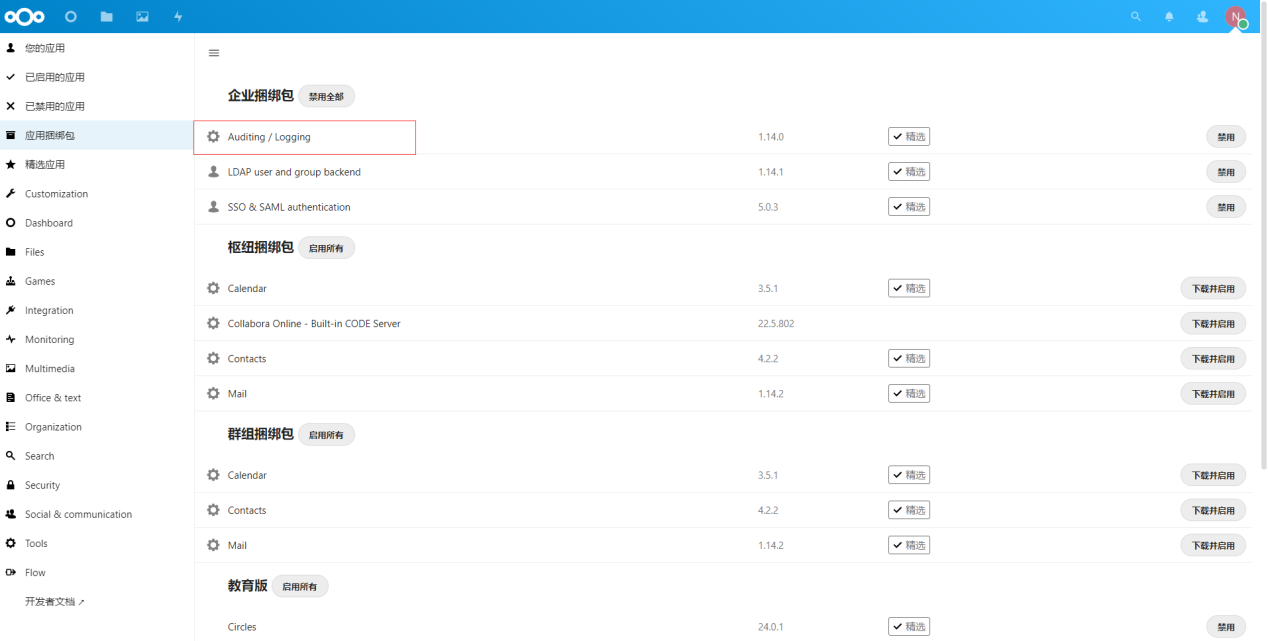
至此，我们已经完成对 NextCloud 的基本配置，重新启动容器后，即可使用管理员账号登录 NextCloud。



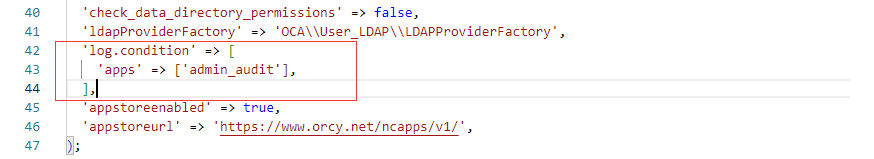
## 软件配置

### 3.1、审计日志

以管理员身份登录后，点击右上角头像，选择“应用”，点击左侧“企业捆绑包”并找到“Auditing/Logging”，确保该选项处于启用状态。



进入 NextCloud 挂载目录，找到 /config/config.php 文件，修改配置如下：



参考文档：

https://docs.nextcloud.com/server/latest/admin\_manual/configuration\_server/logging\_configuration.html#admin-audit-log

### 3.2、LDAP集成

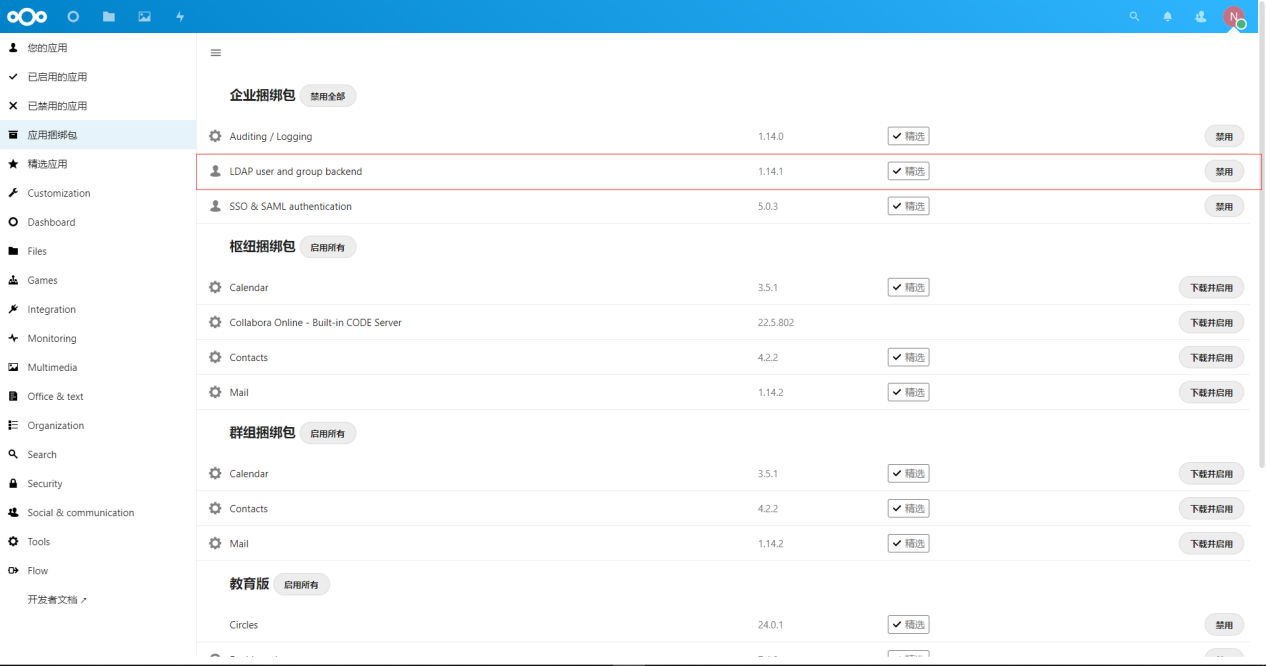
进入 NextCloud 挂载目录，找到 /config/config.php 文件，修改配置如下：



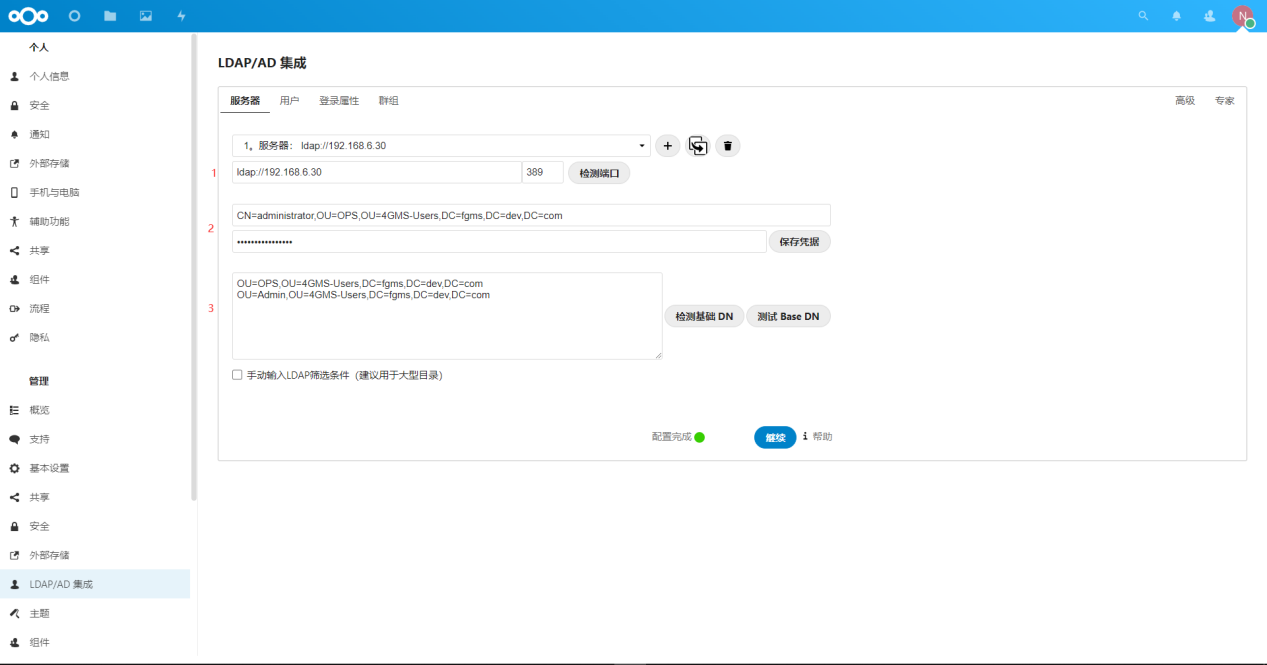
注意，考虑到国内 Github 访问速度非常慢，建议这里使用带加速效果的地址：

<https://www.orcy.net/ncapps/v2>

接下来，以管理员身份登录 NextCloud，点击右上角头像，选择“应用”，点击左侧“企业捆绑包”并找到“LDAP user and group backend”，确保该选项处于启用状态。



点击左上角头像，选择“设置”，在左侧菜单中找到“LDAP/AD集成”，此时，将打开如下界面：



首先，在第一个位置输入域服务的地址，默认使用ldap协议，端口号为389，如果使用 ldaps 协议，请调整为实际的端口号。

其次，在第二个位置分别输入域服务器管理员账号信息。以开发环境为例：

用户DN 为：CN=administrator,OU=OPS,OU=4GMS-Users,DC=fgms,DC=dev,DC=com，

对应的密码为：Stelect2013

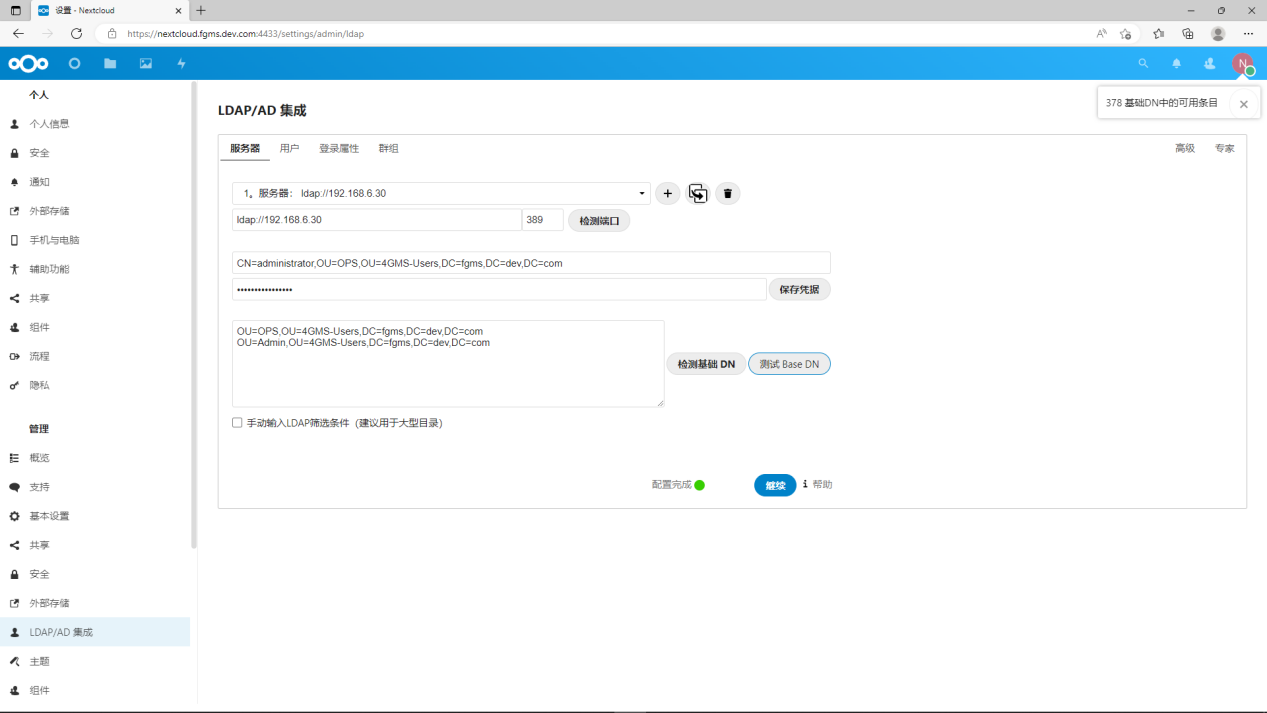
配置时请以实际环境中的域服务器参数为准，配置完成后可以点击“保存凭据”按钮，以验证填入的账号信息是否正确。

然后，在第三个位置输入域服务器的基础 DN 信息。以开发环境为例，每行一个：

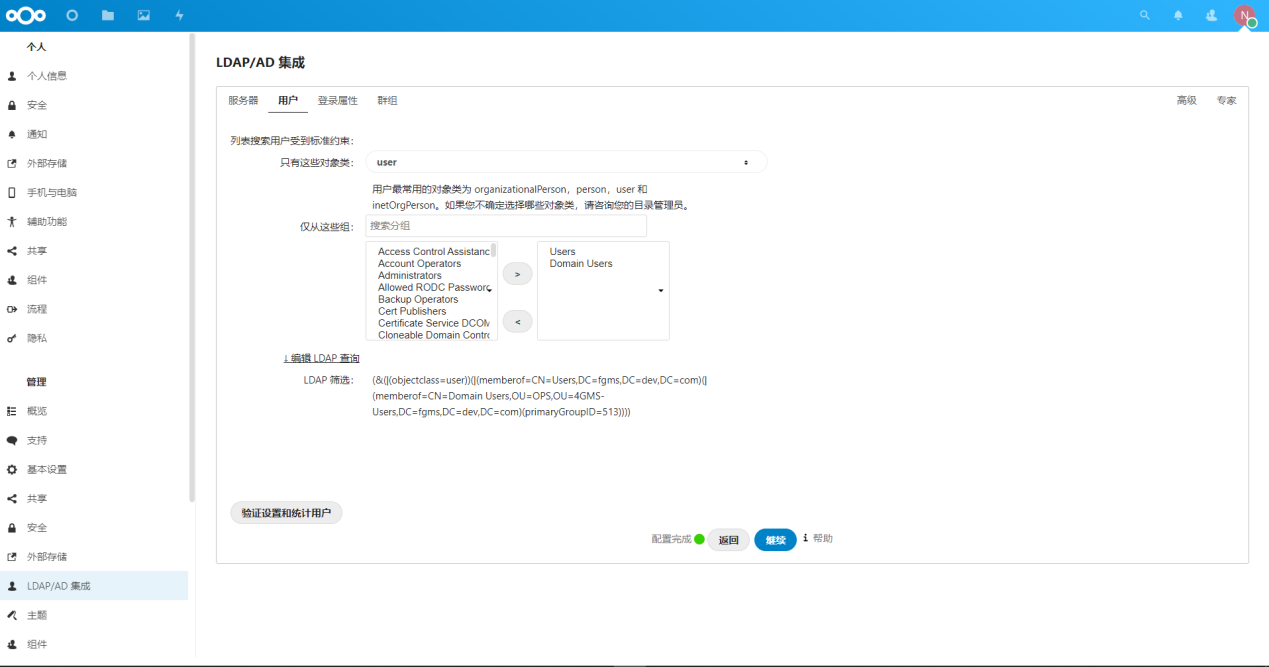
OU=OPS,OU=4GMS-Users,DC=fgms,DC=dev,DC=com

OU=Admin,OU=4GMS-Users,DC=fgms,DC=dev,DC=com

配置时请以实际环境中的域服务器参数为准，配置完成后可以点击“测试 Base DN”按钮，如右上角提示类似下面这样的信息，则表示基础 DN 信息配置正确。

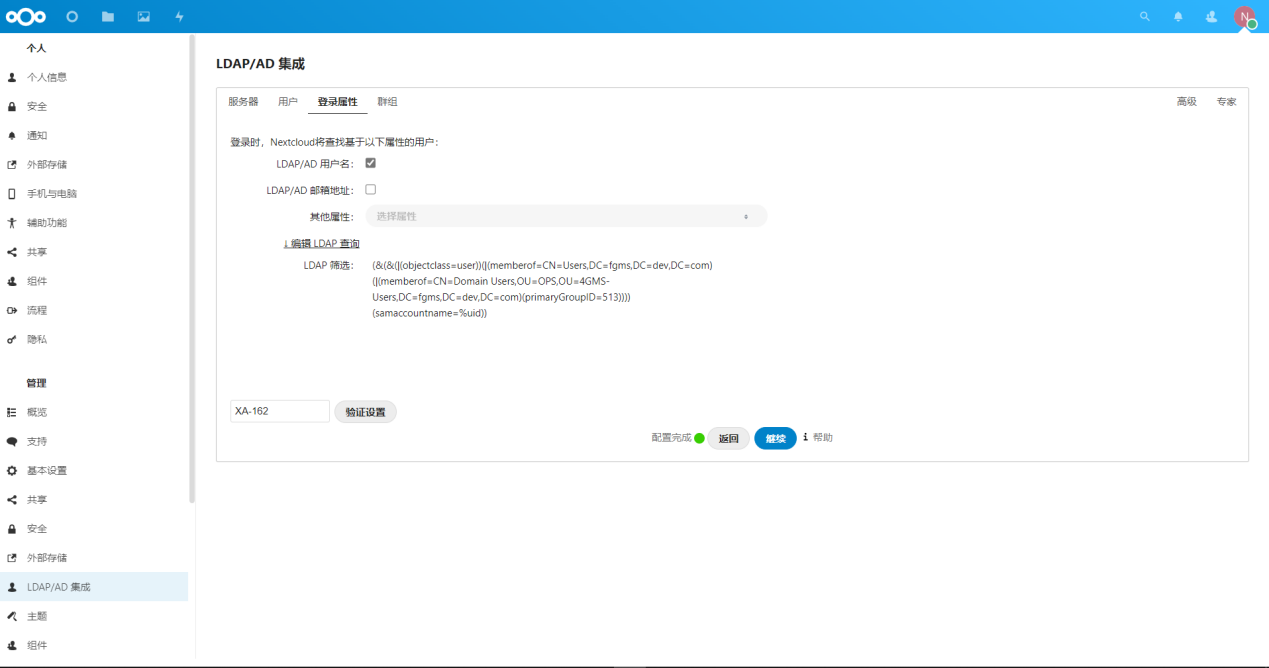


如图所示，当下方显示完成且颜色为绿色时，表示第一步配置完成，点击继续，进入下一步配置。此时，将切换到“用户”选项卡，默认我们只需要 user 这个对象类，如果你有更特殊的需求，可以按自己的实际情况选择合适的对象类。同理，默认我们只需要 Users 和 Domain Users 两个分组，如果你有更特殊的需求，可以按自己的实际情况选择合适的分组，此时，下方会自动带出 LDAP 的筛选条件，你可以点击下方的“验证设置和验证用户”按钮，来检查是否从 AD 筛选到了正确的用户信息。

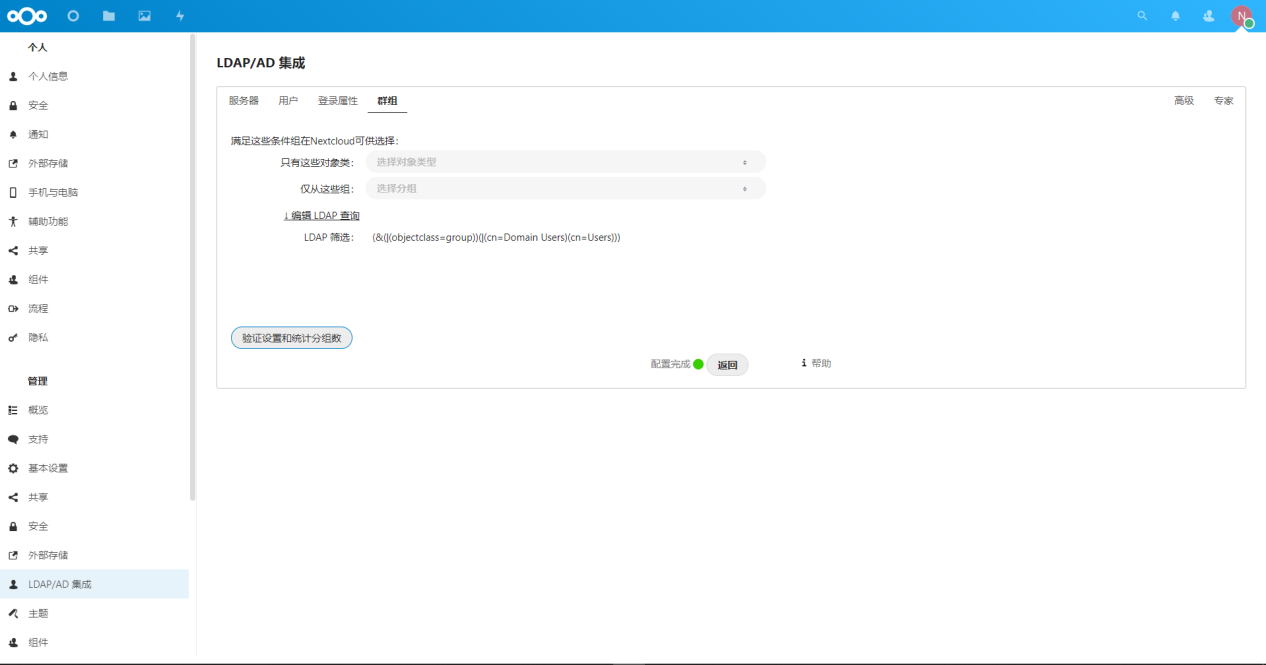


点击“继续”按钮，进入“登录属性”选项卡，勾选 “LDAP/AD用户名”选项。

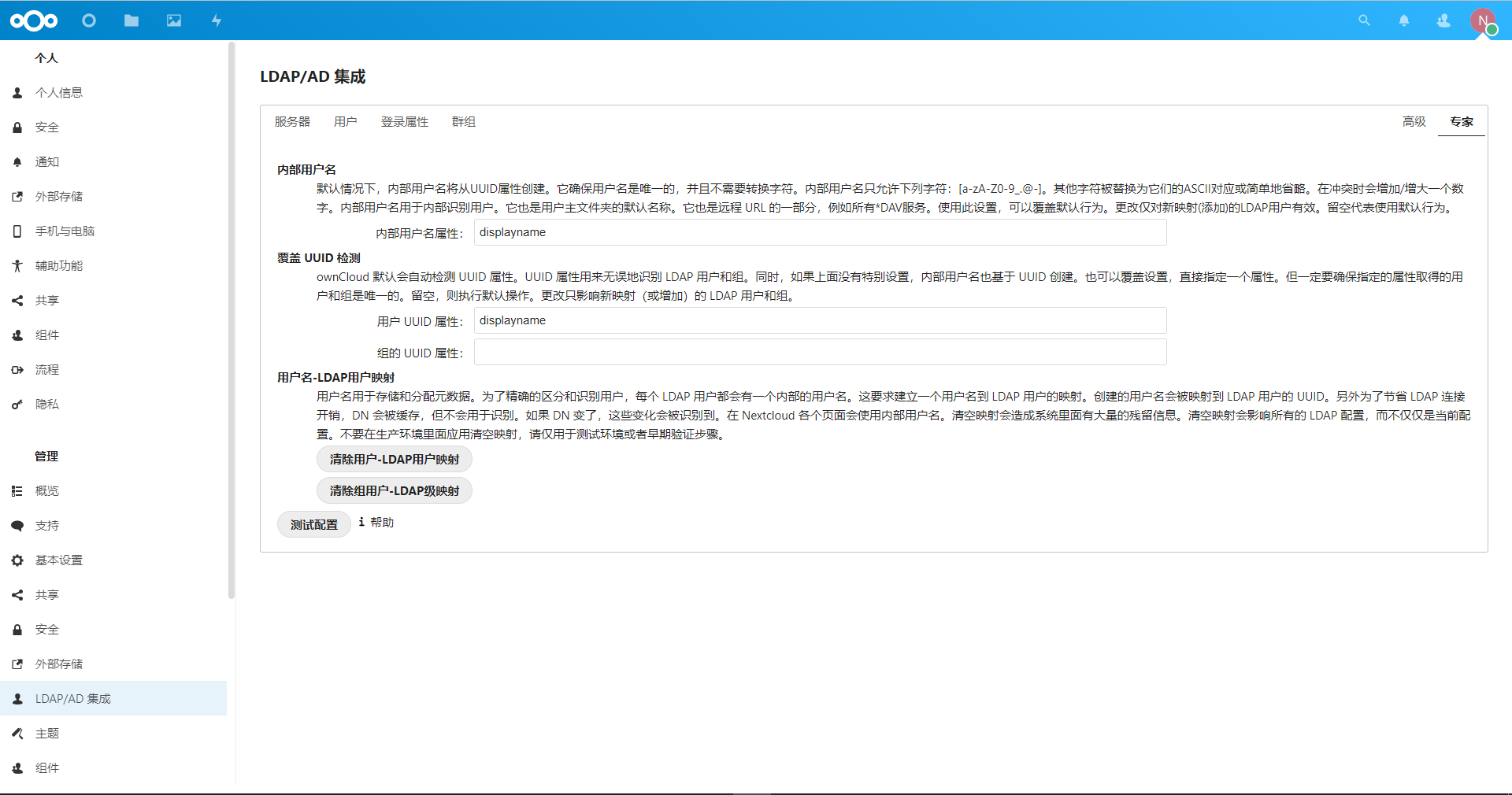
同样地，你可以在下方的输入框中输入一个域账号，来检测当前的配置是否正确：



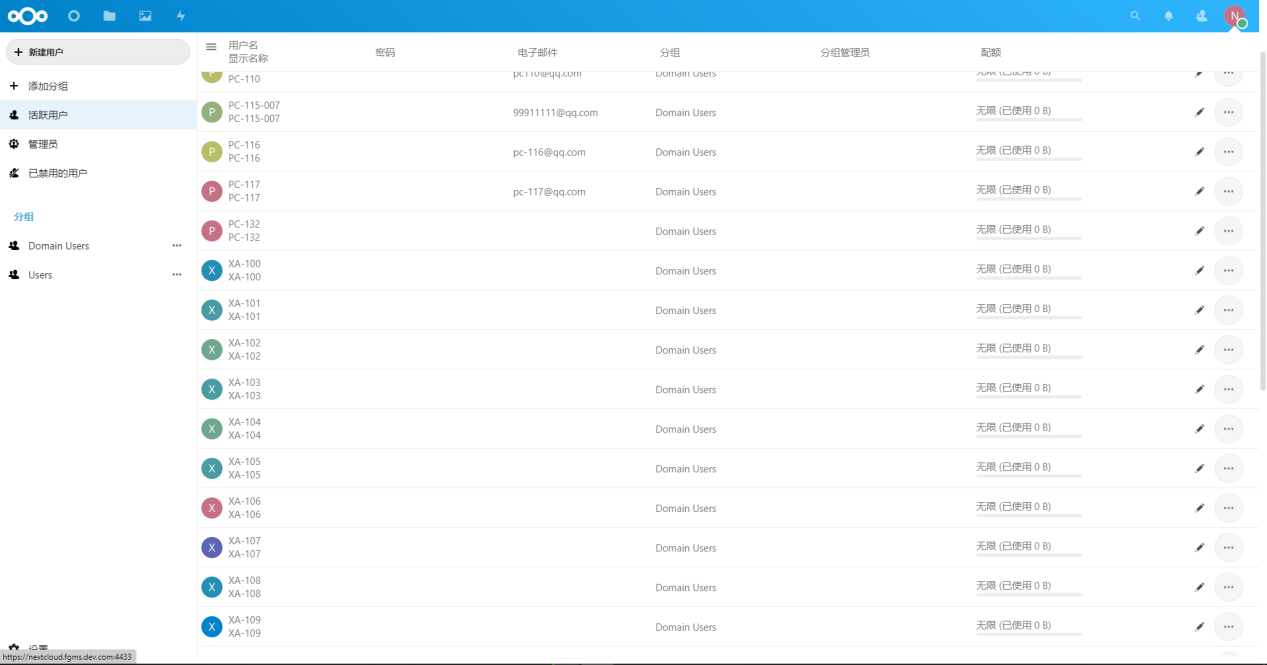
最后，点击“群组”选项卡，同样地，在这里点击“验证设置和统计分组数”按钮，可以验证是否从 AD 筛选到了正确的分组信息。



除此以外，你还需要点击右侧的“专家”打开专家模式，在此模式下，你需要在图中所示的位置填入 displayname，它会保证当你从 AD 中同步用户信息的时候，始终以类似 XA-162 这样的字眼来作为用户的 ID。



一旦完成了上述工作，你可以点击右上角头像，并选择“用户”，在这里你会看到 NextCloud 从 AD 域服务器中同步过来的用户信息，NextCloud 会在后台完成同步工作，需要注意的是，在一个用户没有被同步到 NextCloud 之前，你无法通过该账号登录 NextCloud。

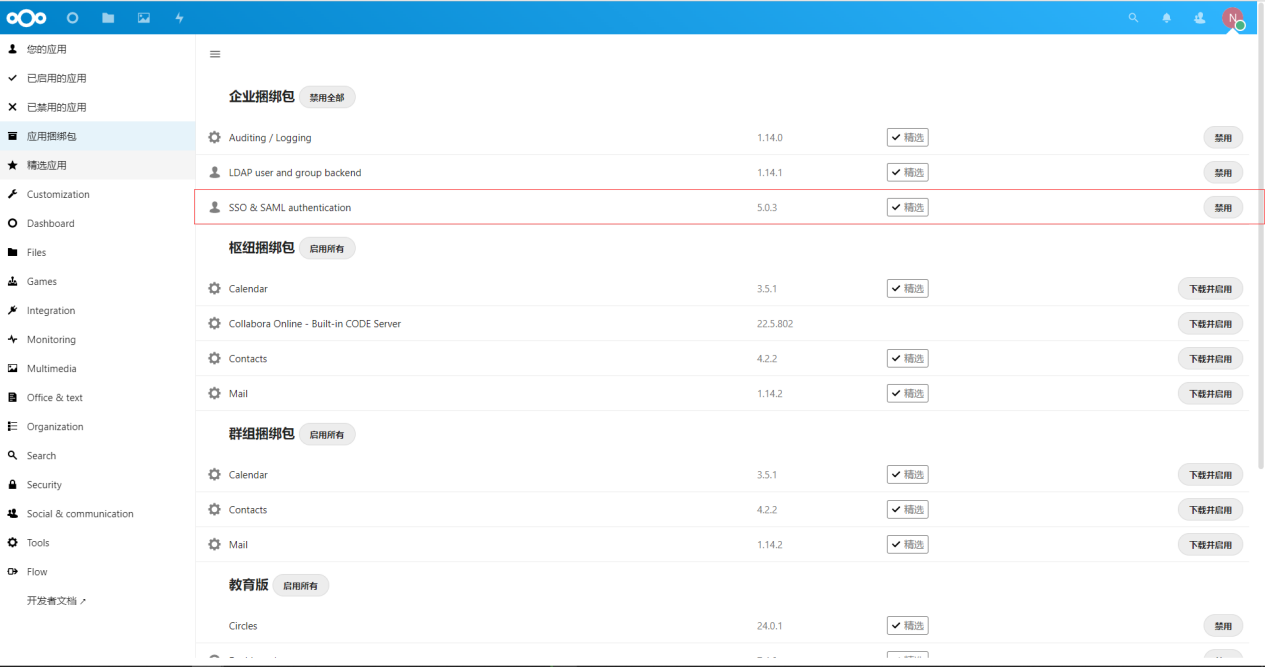


参考链接：

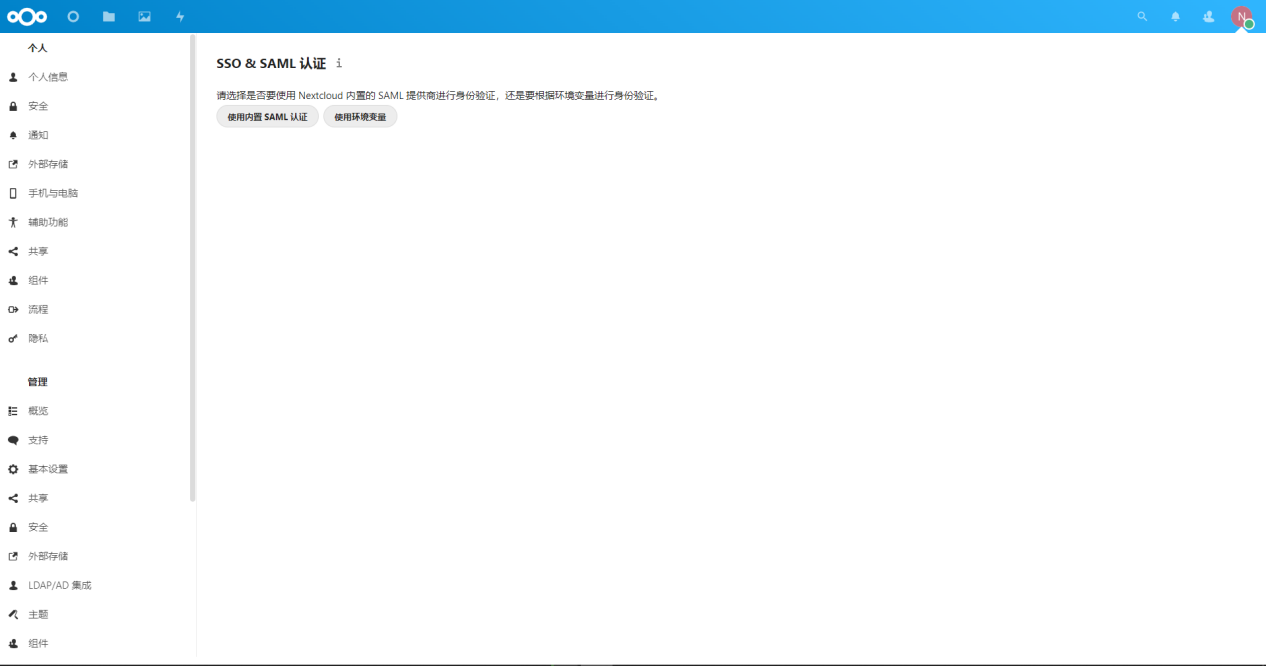
https://docs.nextcloud.com/server/latest/admin\_manual/configuration\_user/user\_auth\_ldap.html

### 3.3 、SSO集成

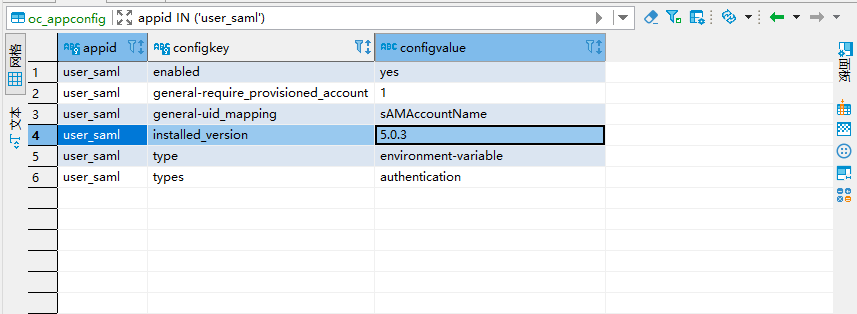
首先，点击右上角头像，选择“应用”，从“企业捆绑包”中找到“SSO & SAML authentication”，首次使用需要下载该扩展模块，因此，我们点击“下载并启用”即可，如下载失败，请多试几次。



等扩展模块安装好以后，在“设置”页面会出现“SSO & SAML 认证”菜单



此时，我们点击使用“使用环境变量”按钮，参数使用默认值即可。此时，打开 NextCloud 的数据库，数据库名、数据库账号取决于第一节的环境变量配置，默认值均为 nextcloud。此刻，需要检查检查 oc\_appconfig 表中是否存在以下数据，如果没有，请手动填入相应的记录。



参考脚本如下：

**INSERT** **INTO** nextcloud.oc\_appconfig

(appid, configkey, configvalue)

**VALUES**('user\_saml', 'type', 'environment-variable');

**INSERT** **INTO** nextcloud.oc\_appconfig

(appid, configkey, configvalue)

**VALUES**('user\_saml', 'general-uid\_mapping', 'sAMAccountName');

**INSERT** **INTO** nextcloud.oc\_appconfig

(appid, configkey, configvalue)

**VALUES**('user\_saml', 'general-require\_provisioned\_account', '1');

接下来，登录域服务器，在命令行中输入以下命令：

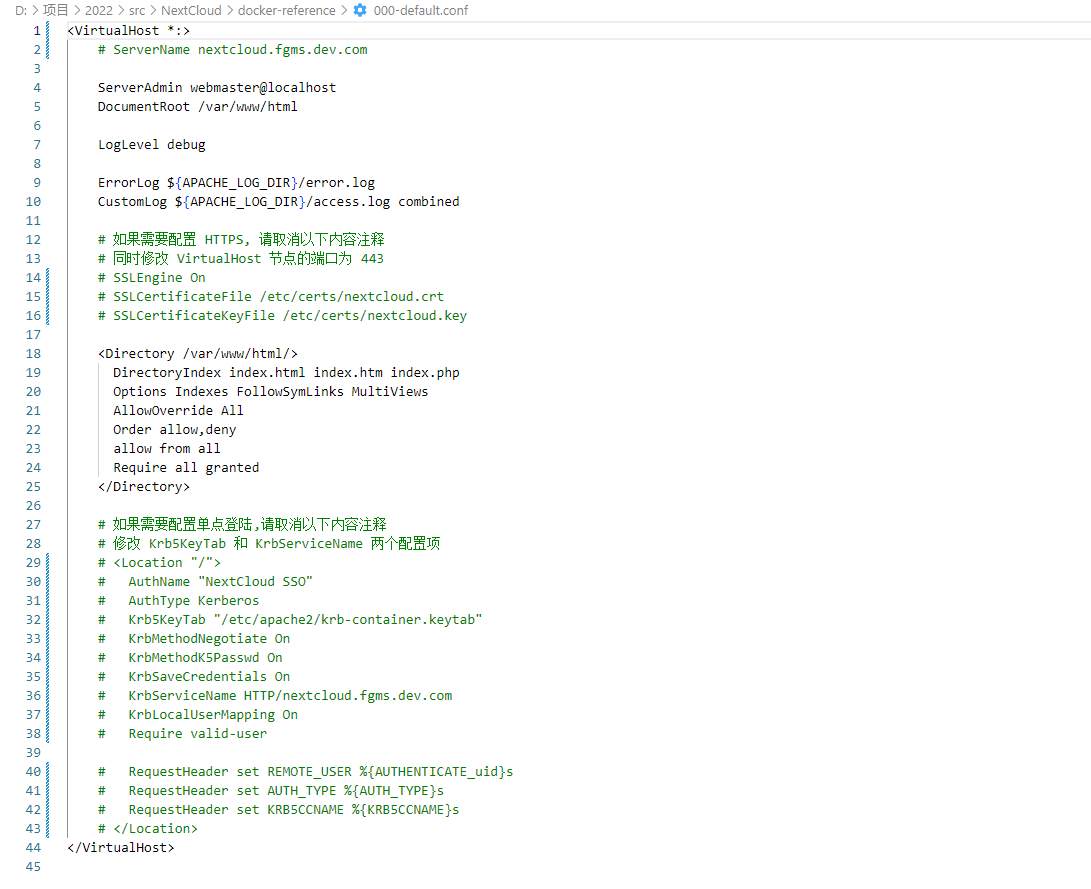
ktpass -out krb-container.keytab -mapUser <Krb-User>@FGMS.DEV.COM /pass <Krb-User-Pass> -ptype KRB5\_NT\_PRINCIPAL -princ HTTP/<NextCloud-Server-Name>@FGMS.DEV.COM -crypto ALL

其中：

* <NextCloud-Server-Name> 是在环境准备阶段我们为NextCloud 分配的域名，这里假设为 nextcloud.fgms.dev.com。
* <Krb-User> 是我们在域服务器上为 NextCloud 创建的一个域账号，<Krb-User-Pass> 是该域账号对应的密码。

切记，这里不能使用管理员账号，因为管理员账号往往绑定着某个 SPN，会和我们这里的 SPN 发生冲突。

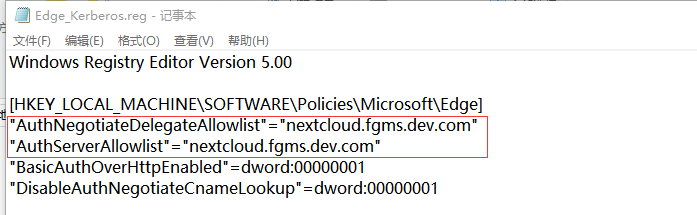
此时，我们会得到一个名为 krb-container.keytab 的文件，请将此文件复制到 docker-reference 目录下，并修改该目录下的 000-default.conf 文件：



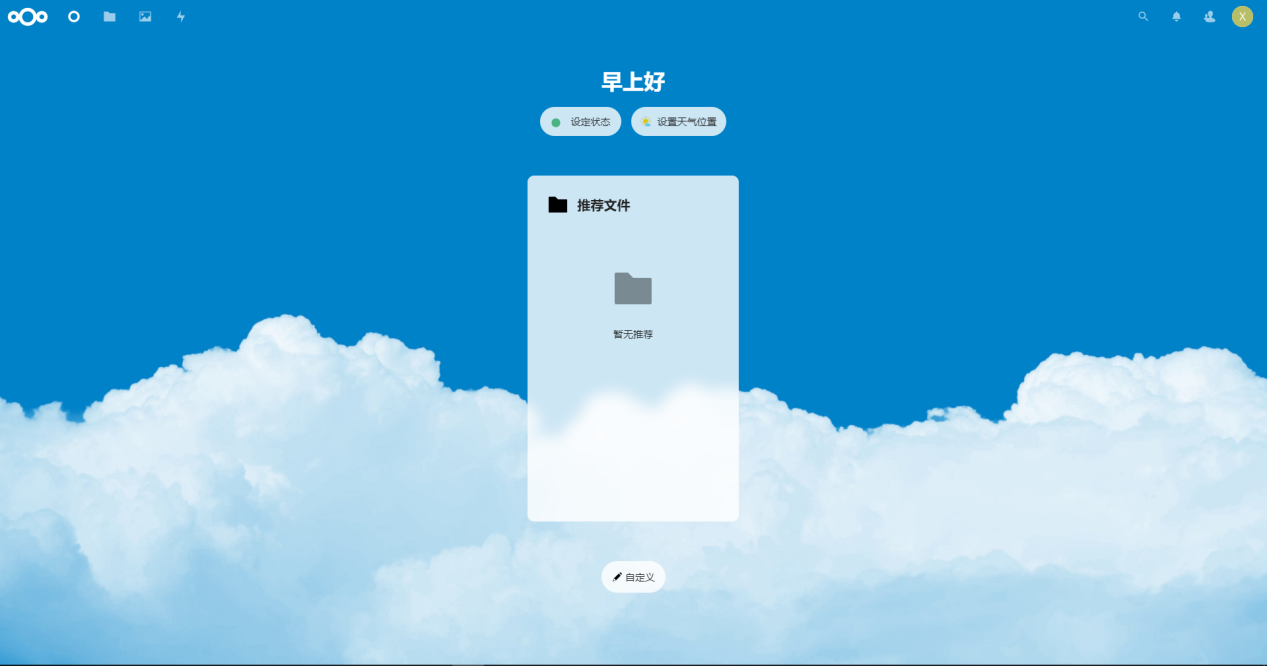
* 请取消第2行的注释，域名以实际分配的域名为准，域名不含端口号或者协议，且该值与生成keytab 文件时指定的 SPN 的域名必须一致。
* 请取消第14行到第16行的注释，如果你希望配置 HTTPS，与此同时，你应该修改监听的端口为 443，此后只能通过 HTTPS 协议访问 NextCloud
* 请取消第29行到第43行的注释，同时修改第36行的域名为实际的域名，理论上该值应该与第2行一致

其中，最重要的部分是 Location 节点，它表示对根路径 “/” 下的文件启用 Kerberos 认证，请参考实际环境中的域名，修改 ServerName 和 KrbServiceName 两个参数，注意保留前面的 HTTP/ 前缀。配置完成后，需要重启容器，确保 Aapche 能正确应用以上配置。

除了以上的后端设置，你还需要针对浏览器做一点设置，请执行 Edge\_Kerberos.reg 脚本，以确保浏览器可以将本地认证信息发送给 Kerberos，该脚本内容如下：



请确保脚本中的的AuthNegotiateDelegateAllowlist 和 AuthServerAllowlist 两个键维护了正确的域名，该值必须与上面提到的域名一致，因为Kerberos 利用这个域名来判断是否需要发送认证信息给 Apache，请注意，使用单点登陆的前提是当前设备加入了 AD 域。



至此，基于 Kerberos 的单点登陆就配置完成了。重启容器后，用户可以在不输入密码的情况下直接打开 NextCloud。

### 3.3 、给文档增加水印

增加水印依赖于 pdftk 和 ps2pdf 两个 Linux 工具。目前，已在 Dockerfile 中集成，我们可以通过 /docker-reference/workflow/add-watermark.sh 这个脚本来对指定的 PDF 文件增加水印。在此之前，你需要安装 workflow\_script 扩展应用。首先，从以下地址下载程序包：

<https://apps.nextcloud.com/apps/workflow_script/releases?platform=24>

请注意，当前的 NextCloud 的大版本号为24，安装时请务必确保 NextCloud 版本与 workflow\_script 版本匹配。下载完成后，将其解压到 /custom\_apps/workflow\_script/ 目录下。接下来，进入 NextCloud 的数据库中，执行以下脚本，完成 workflow\_script 的安装：

INSERT INTO nextcloud.oc\_appconfig

(appid, configkey, configvalue)

VALUES('workflow\_script', 'enabled', 'yes');

INSERT INTO nextcloud.oc\_appconfig

(appid, configkey, configvalue)

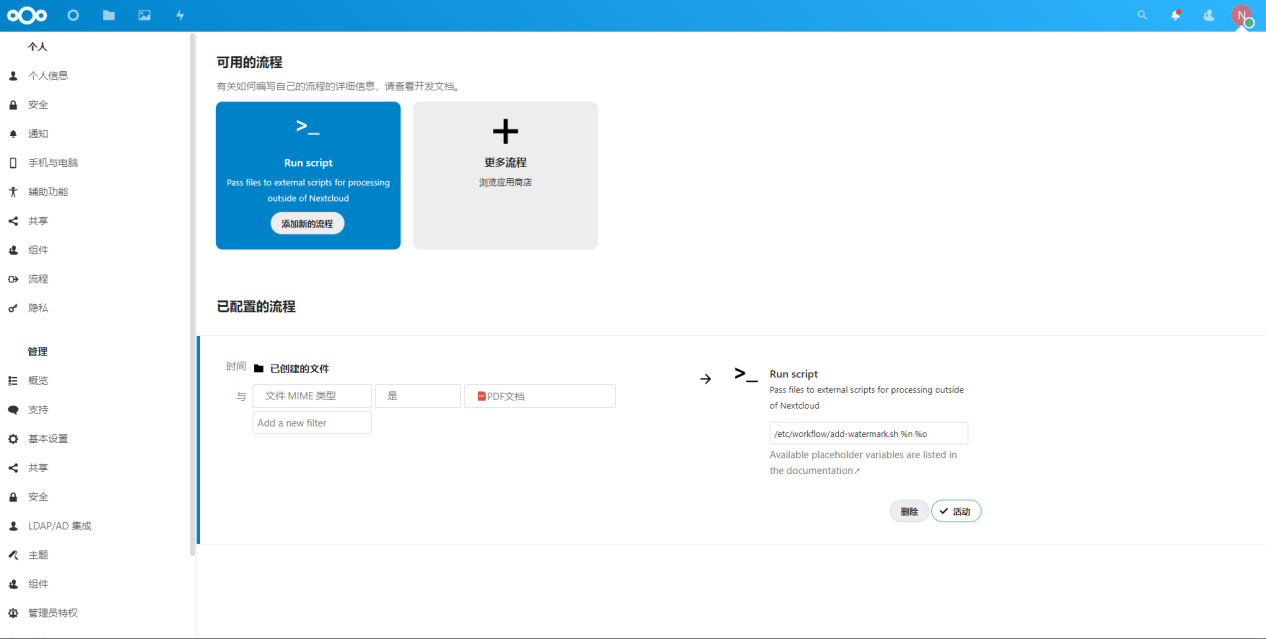
VALUES('workflow\_script', 'installed\_version', '1.9.0');

INSERT INTO nextcloud.oc\_appconfig

(appid, configkey, configvalue)

VALUES('workflow\_script', 'types', 'filesystem');

安装后，以管理员身份登录 NextCloud，从左侧菜单中选择“流程”，新建一个针对 PDF 文件的流程：



此时，如果 NextCloud 的定时任务 (cron.php) 正确执行，则可以在上传完一个 PDF 文档后为文件增加水印效果，如下图所示：

