

Docker 问题与方法
Docker 数据迁移
Docker 资源清理
Docker 配置代理和私有仓库
Docker 使用普通用户操作
无损升级Docker容器的镜像版本
00 LearnDocker
Deploy
Cloudreve 个人网盘
filebrowser 个人网盘
Gerrit 代码评审工具
Gitea+Drone DevOps环境
Harbor 云原生制品库
LDAP 认证服务
miniserve 文件服务器
Navidrome 音乐服务
OwnCloud 企业级网盘
docker-registry-ui 管理镜像
TeamCity JetBrains CI服务

Gerrit 代码评审工具

概要: Gerrit Code Review是一个流行的代码评审工具，也是Git代码仓库。本文介绍使用Docker的方式部署Gerrit。

准备工作

文件结构

Text Only

```

1 $ tree
2 .
3   └── docker-compose.yaml
4   └── gerrit/
5     ├── etc/
6     |   └── gerrit.config
7     |   └── secure.config
8   └── ldap/
9
10  3 directories, 3 files

```

docker-compose.yaml

YAML

```

1 # https://hub.docker.com/r/gerritcodereview/gerrit
2 version: '3'
3
4 services:
5   gerrit:
6     container_name: gerrit_server
7     image: gerritcodereview/gerrit
8     user: root
9     ports:
10       - "29418:29418"
11       - "80:8080"
12     depends_on:
13       - ldap
14     volumes:
15       - ./gerrit/etc:/var/gerrit/etc
16       - ./gerrit/git:/var/gerrit/git
17       - ./gerrit/db:/var/gerrit/db
18       - ./gerrit/index:/var/gerrit/index
19       - ./gerrit/cache:/var/gerrit/cache
20     environment:
21       - CANONICAL_WEB_URL=http://192.168.2.111:80
22     # command: init # 首次执行必须取消此行注释
23
24
25   ldap:
26     container_name: gerrit_opendap
27     image: osixia/opendap
28     restart: always
29     ports:
30       - "389:389"
31       - "636:636"
32     environment:
33       - LDAP_ADMIN_PASSWORD=ldap_passwd # 与secure.config一致
34     volumes:
35       - ./ldap/var:/var/lib/ldap
36       - ./ldap/etc:/etc/ldap/slapd.d
37
38   ldap-admin:
39     container_name: gerrit_phpldapadmin
40     image: osixia/phpldapadmin
41     restart: always
42     ports:
43       - "443:443"
44     environment:
45       - PHPLDAPADMIN_LDAP_HOSTS=ldap

```

目录

准备工作
文件结构
docker-compose.yaml
gerrit.config
secure.config
首次启动 (初始化)
正常启动
配置LDAP帐户
登录LDAP管理网页
新建组织: gerrit
新建用户组: gerrit_admin
新建用户: lzwang
创建完成
登录gerrit
参考

提示

1. 需要写入gerrit的配置文件 `gerrit.config`
2. 需要写入gerrit的登录配置文件 `secure.config`
3. 首次启动需要取消注释上面的 `# command: init` 这一行

gerrit.config

Properties

```
1 [gerrit]
2   basePath = git
3
4 [index]
5   type = LUCENE
6
7 [auth]
8   type = ldap
9   gitBasicAuth = true
10
11 [ldap]
12   server = ldap://ldap
13   username=cn=admin,dc=example,dc=org
14   accountBase = dc=example,dc=org
15   accountPattern = (&(objectClass=person)(uid=${username}))
16   accountFullName = displayName
17   accountEmailAddress = mail
18
19 [sendemail]
20   smtpServer = localhost
21
22 [sshd]
23   listenAddress = *:29418
24
25 [httpd]
26   listenUrl = http://*:8080/
27
28 [cache]
29   directory = cache
30
31 [container]
32   user = root
```

secure.config

Properties

```
1 [ldap]
2   password = ldap_passwd
```

首次启动（初始化）

将 `docker-compose.yaml` 的 `# command: init` 这一行取消注释，然后在项目根目录执行

Bash

```
1 docker-compose up gerrit
```

接着，gerrit服务会自动进行初始化，等到初始化完毕，docker容器会自动停止

Bash

```
1 gerrit_server exited with code 0
```

正常启动

将 `docker-compose.yaml` 的 `command: init` 这一行注释掉，然后在项目根目录执行

Bash

```
1 docker-compose up -d
```

此时所有服务都应该正常启动，gerrit的服务会有日志输出

Text Only

1 | Gerrit Code Review 3.x.x ready

配置LDAP帐户

登录LDAP管理网页

在成功启动服务后，由于gerrit的授权由LDAP提供，所以首先需要登录到 `ldap-admin` 网页去新建一个帐户用于登录gerrit。打开`https://192.168.2.111:443`网页（注意必须是 https）：



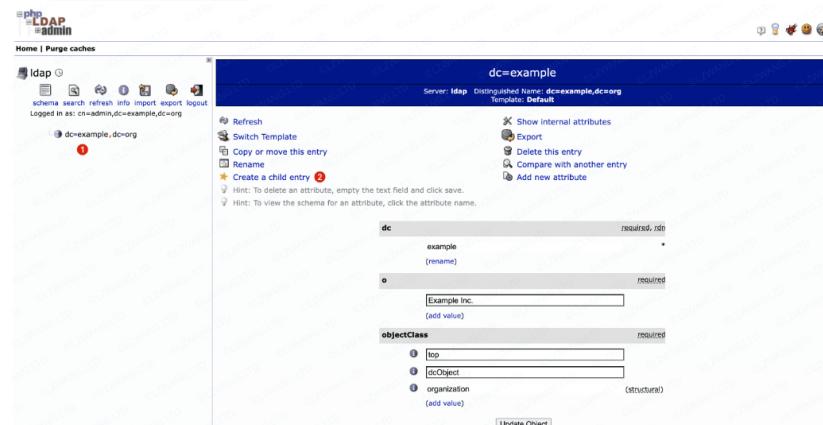
其中

- Login DN 填写: `cn=admin,dc=example,dc=org`
- Password 填写: `ldap_passwd`

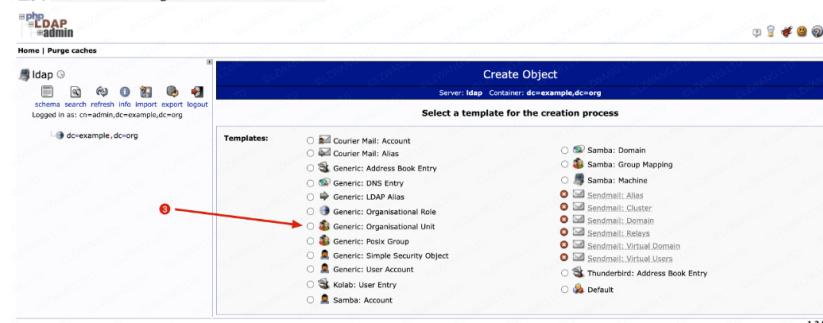
点击Authenticate即可登录LDAP管理页面。

新建组织: gerrit

点击 Create a child entry



选择 Generic: Organisational Unit



填入组织名 `gerrit`，并创建对象



新建用户组: gerrit admin

选择我们刚刚建好的 ou=gerrit , 然后点击 Create a child entry

The screenshot shows the 'Create a child entry' page for the 'ou=gerrit' object. The 'Template' dropdown is set to 'Default'. The 'objectClass' section contains 'top' and 'organizationalUnit'. The 'cn' field is populated with 'gerrit'. The 'gidNumber' field is set to '500'. The 'Update Object' button is at the bottom right.

选择 Generic: Posix Group

The screenshot shows the 'Select a template for the creation process' dialog. The 'Templates:' list includes various options like 'Courier Mail: Account', 'Generic: Posix Group', and 'Samba: Domain'. A red arrow points from the 'Templates:' list to the 'Generic: Posix Group' option, which is highlighted.

填入用户组名 gerrit_admin , 并创建对象

The screenshot shows the 'Create Object' dialog for 'New Posix Group (Step 1 of 1)'. The 'Template' is set to 'Generic: Posix Group (posixGroup)'. The 'Group' field is filled with 'gerrit_admin'. The 'GID Number' field is set to '500'. The 'Users' field is empty. The 'Create Object' button is at the bottom right.

新建用户: lzwang

选择我们刚刚建好的 cn=gerrit_admin , 然后点击 Create a child entry

The screenshot shows the 'Create a child entry' page for the 'cn=gerrit_admin' object. The 'Template' dropdown is set to 'Default'. The 'objectClass' section contains 'top' and 'posixAccount'. The 'cn' field is populated with 'gerrit_admin'. The 'gidNumber' field is set to '500'. The 'Update Object' button is at the bottom right.

选择 Generic: User Account

The screenshot shows the 'Select a template for the creation process' dialog. The 'Templates:' list includes various options like 'Courier Mail: Account', 'Generic: Address Book Entry', and 'Samba: Domain'. A red arrow points from the 'Templates:' list to the 'Generic: User Account' option, which is highlighted.



填入如图所示的信息后，创建此对象

创建完成

至此，我们的第一个LDAP帐户创建完毕（吐槽：真费劲...）。

登录gerrit

打开网页<http://192.168.2.111:80>，使用上面LDAP创建的帐户密码即可登录

参考

- [gerritcodereview/gerrit - Docker Image | Docker Hub](#)