



全球运维大会

2016

DevOps 2.0: 重塑运维价值



北京站

会议时间：12月16日 - 12月17日

会议地点：北京国际会议中心

主办单位：



车联网的统一身份认证系统演变过程

许颖维 运维总监



目录

1 选择

2 需求分析

3 如何实施

4 演变历程

5 总结



没有万能的系统

- 1 统一认证：一个用户只有一个账号密码；
- 2 生效快捷：用户信息变更之后快速在所有服务器上生效；
- 3 最简场景：ssh & scp；



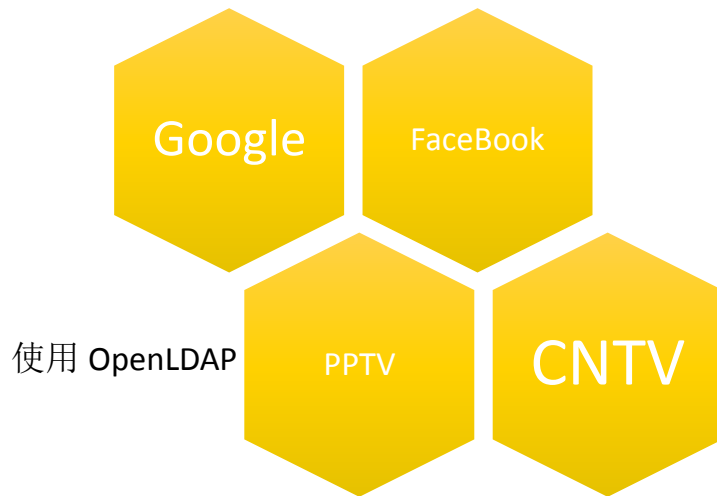
选择

开源

- Kerberos
- OpenLDAP
-

商业

- AD
- 堡垒机
-

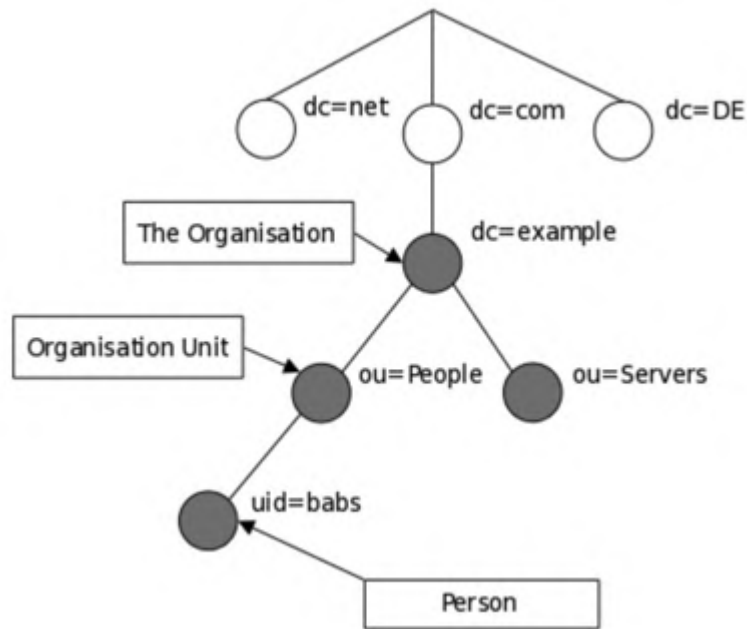


商业与开源对比

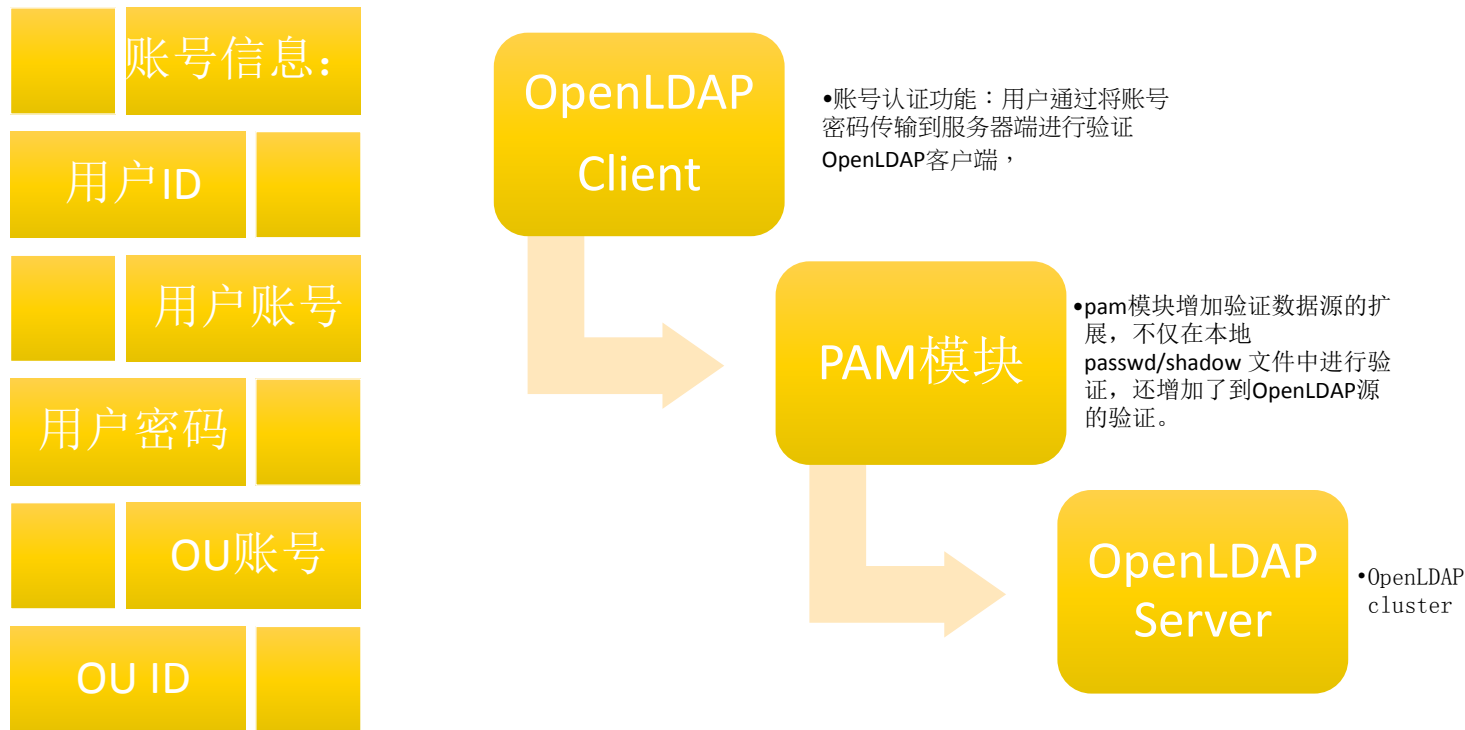


OpenLDAP介绍

- OpenLDAP Software is an open source implementation of the Lightweight Directory Access Protocol.
- How does LDAP work
- X.500
- DB : BerKeley DB



分析：用户认证过程



分析：用户提权过程

提权功能

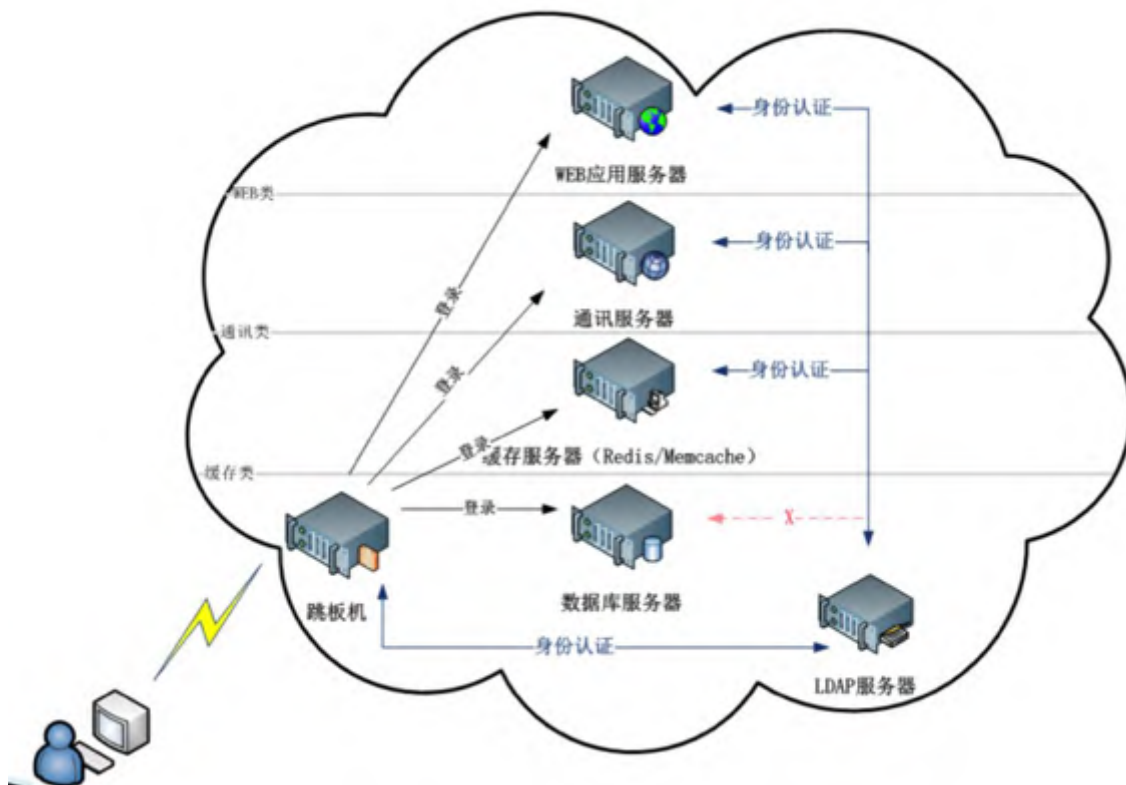
- 用户通过个人账号登录服务器之后，可以使用sudo来进行提权

权限定义

- 权限定义分为默认权限和临时权限，默认权限是在OpenLDAP中定义，默认所有用户的权限都是：
- `sudo su - root`



分析：用户使用过程



安装部署 (CentOS 5.8)

Server



- ☐ `openldap-devel-$VERSION`
- ☐ `openldap-$VERSION`
- ☐ `openldap-servers-$VERSION`
- ☐ `openldap24-libs-$VERSION`
- ☐ `openldap-servers-sql-$VERSION`

Client

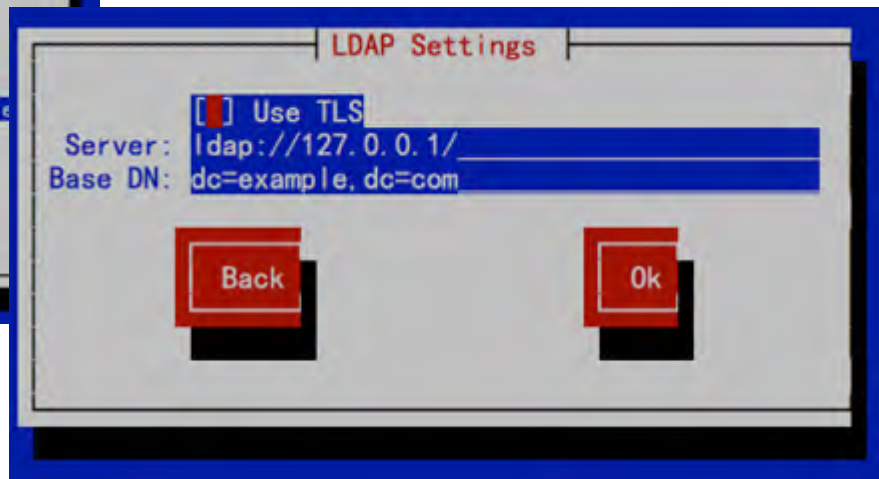
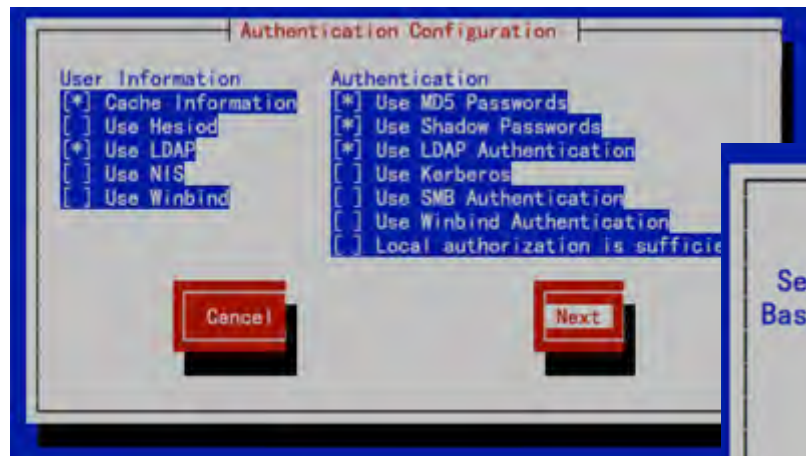


- ☐ `pam_ldap-$VERSION`
- ☐ `nss-pam-ldapd-$VERSION`
- ☐ `openldap-$VERSION`
- ☐ `openldap-clients-$VERSION`



客户端配置 (CentOS 5.8)

- Command: Authconfig-tui



客户端配置（注意事项）

- ConfigureFile : /etc/ldap.conf

```
URI ldaps://[redacted]  
BASE dc=[redacted], dc=com, dc=cn  
TLS_CACERTDIR /etc/openldap/cacerts  
  
ssl start_tls  
ssl on  
  
tls_cacertfile /etc/openldap/cacerts/cacert.pem  
tls_reqcert demand  
  
pam_password md5  
  
sudoers_base ou=SUDOers, dc=[redacted], dc=com, dc=cn  
  
timelimit 120  
  
bind_timelimit 120  
  
idle_timelimit 3600
```



客户端配置（注意事项）

- ConfigureFile : /etc/nsswitch.conf

```
#group:      db files nisplus nis
passwd:      files ldap
shadow:      files ldap
group:       files ldap
sudoers:     files ldap
```

```
services:    files
netgroup:    files ldap
publickey:   nisplus
automount:   files ldap
aliases:     files nisplus
```



客户端配置（注意事项）

- ConfigureFile : /etc/sysconfig/authconfig

```
[root@localhost ~]# cat /etc/sysconfig/authconfig
USEWINBINDAUTH=no
USEKERBEROS=no
USESYSNETAUTH=no
USEPAMACCESS=no
USEMKHOMEDIR=no
FORCESMARTCARD=no
USESMBAUTH=no
USESMBARTCARD=no
USELDAPAUTH=yes
USEDDB=no
USEWINBIND=no
USESHADOW=yes
PASSWDALGORITHM=md5
USELOCALAUTHORIZE=yes
USEPASSWDQC=no
USELDAP=yes
USEHESIOD=no
USECRACKLIB=yes
USENIS=no
```



客户端配置（注意事项）

- ConfigureFile：/etc/pam.d/system-auth

```
[root@localhost ~]# cat /etc/pam.d/system-auth
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth      required      pam_env.so
auth      sufficient    pam_unix.so nullok try_first_pass
auth      requisite     pam_succeed_if.so uid >= 500 quiet
auth      sufficient    pam_ldap.so use_first_pass
auth      required      pam_deny.so

account   required      pam_unix.so broken_shadow
account   sufficient    pam_localuser.so
account   sufficient    pam_succeed_if.so uid < 500 quiet
account   [default=bad success=ok user_unknown=ignore] pam_ldap.so
account   required      pam_permit.so

password  requisite     pam_cracklib.so try_first_pass retry=3 dcredit=-1 ocredit=-1
password  sufficient    pam_unix.so md5 shadow nullok try_first_pass use_authtok
password  sufficient    pam_ldap.so use_authtok
password  required      pam_deny.so

session   optional     pam_keyinit.so revoke
session   required      pam_limits.so
session   [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session   required      /lib/security/$ISA/pam_mkhome.so skel=/etc/skel umask=0077
session   required      pam_unix.so
session   optional     pam_ldap.so
```



演变的开始：权限的精细化管理

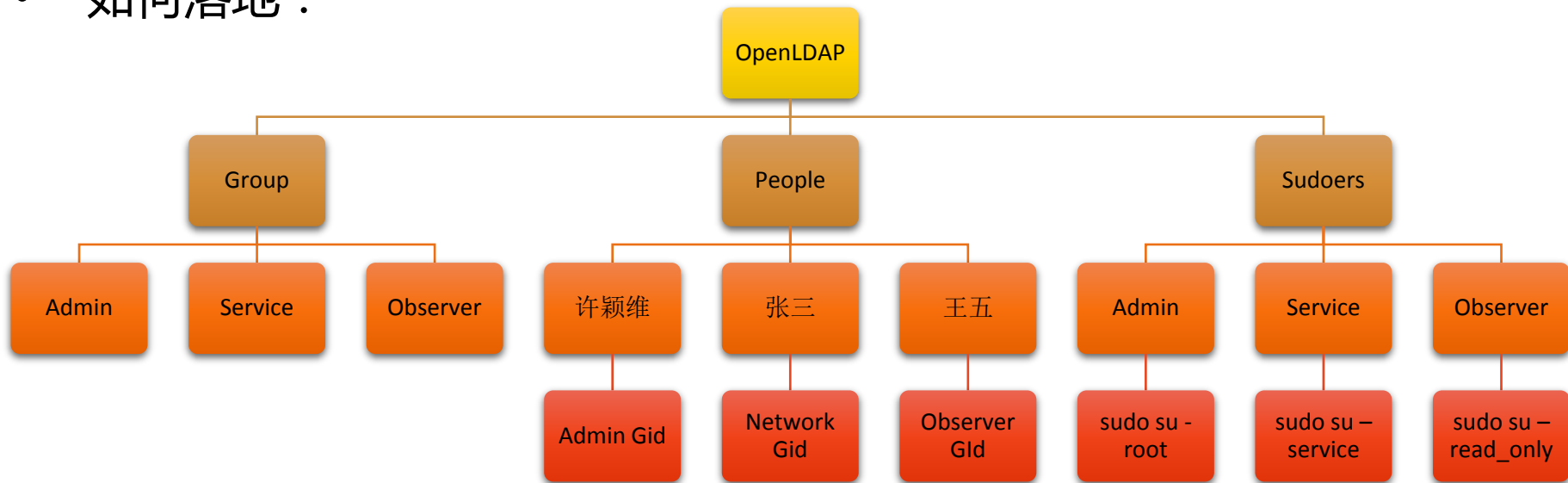
- 如何细化：

admin	•sudo su - root
service	•sudo su – oracle/mysql •sudo su – service
Network	•sudo su – Network-command
observer	•sudo su – read_only
personal	•none



权限：精细化管理

- 如何落地：



权限：精细化管理

- 如何实现：（ UserInformation、 GroupInformation ）

```
# dev, Group, [redacted] com.cn
dn: cn=dev,ou=Group,dc=[redacted],dc=com,dc=cn
objectClass: posixGroup
objectClass: top
cn: dev
gidNumber: 200100

# ops, Group, [redacted] com.cn
dn: cn=ops,ou=Group,dc=[redacted],dc=com,dc=cn
objectClass: posixGroup
objectClass: top
cn: ops
gidNumber: 200101
```

```
xuyingwei, People, [redacted] com.cn
dn: uid=xuyingwei,ou=People,dc=[redacted],dc=com,dc=cn
cn: xuyingwei
homeDirectory: /home/xuyingwei
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
objectClass: top
o: cn=ldapmanager,dc=[redacted],dc=com,dc=cn
shadowMax: 99999
uidNumber: 200101
uid: xuyingwei
gidNumber: 200102
```



权限：精细化管理

- 如何实现：(SudoersInformation)

```
■ SUDOers, [redacted] com.cn
dn: ou=SUDOers, dc=[redacted], dc=com, dc=cn
objectClass: top
objectClass: organizationalUnit
ou: SUDOers

■ defaults, SUDOers, [redacted] com.cn
dn: cn=defaults, ou=SUDOers, dc=[redacted], dc=com, dc=cn
cn: defaults
sudoOption: ignore_dot
sudoOption: !mail_no_user
sudoOption: !root_sudo
sudoOption: log_host
sudoOption: logfile=/var/log/sudolog
sudoOption: !syslog
sudoOption: timestamp_timeout=10
objectClass: top
objectClass: sudoRole
description: DefaultsudoOption's
```

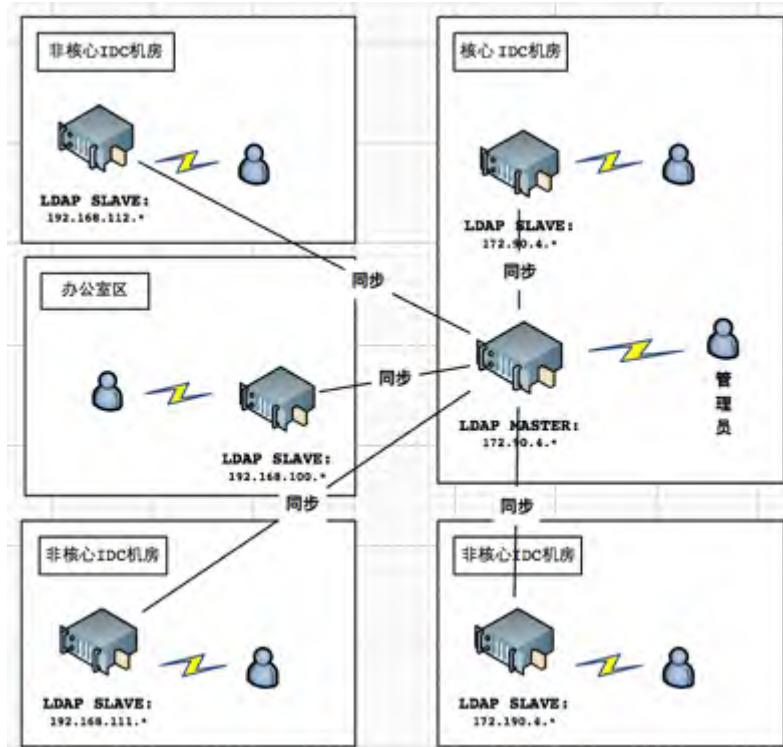
```
■ %dev, SUDOers, [redacted] com.cn
dn: cn=%dev, ou=SUDOers, dc=[redacted], dc=com, dc=cn
cn: %dev
objectClass: sudoRole
objectClass: top
sudoUser: %dev
sudoCommand: /bin/su - yanfa_ro
sudoHost: ALL

■ %ops, SUDOers, [redacted] com.cn
dn: cn=%ops, ou=SUDOers, dc=[redacted], dc=com, dc=cn
cn: %ops
objectClass: sudoRole
objectClass: top
sudoUser: %ops
sudoCommand: /bin/su - lbs
sudoCommand: /bin/su - padm
sudoHost: ALL
```



演变的继续：分布式管理

1. 主从关系：一主多从
2. 只读过程：保持不变
3. 读写过程：读写分离



演变的继续：分布式管理

- Master :

```
Replicas of this database
replogfile /var/lib/ldap/openldap-master-replog
replica host=ldap-1.example.com:389 starttls=critical
        bindmethod=sasl saslmech=GSSAPI
        authcid=host/ldap-master.example.com@EXAMPLE.COM
replogfile /var/lib/ldap/replog

replica host=192.168.1.100:389
        binddn="cn=ldap-master,dc=com,dc=cn"
        credentials=secret
        bindmethod=sasl

replica host=192.168.1.101:389
        binddn="cn=ldap-master,dc=com,dc=cn"
        credentials=secret
        bindmethod=sasl

replica host=192.168.1.102:389
        binddn="cn=ldap-master,dc=com,dc=cn"
        credentials=secret
        bindmethod=sasl
```

Slave:

```
Replicas of this database
replogfile /var/lib/ldap/openldap-master-replog
replica host=ldap-1.example.com:389 starttls=critical
        bindmethod=sasl saslmech=GSSAPI
        authcid=host/ldap-master.example.com@EXAMPLE.COM
replogfile /var/lib/ldap/replog

ldap. ops. 192.168.1.100, cn => 192.168.1.100
updatedn "cn=ldap-master,dc=com,dc=cn"
updateref ldaps://ldap. ops. 192.168.1.100, cn:636
```



演变的继续：分布式管理

- Slurpd.replog

```
replica: [REDACTED]
replica: [REDACTED]
replica: [REDACTED]
replica: [REDACTED]
replica: [REDACTED]
time: 1472541601
dn: uid=[REDACTED], ou=People, dc=[REDACTED], dc=com, dc=cn
changetype: modify
replace: userPassword
userPassword:: e1NTSEF9S21EYXBHUKIFMG1tQzVDUnZuSFFuY20wY1diZFZDbDE=
-
replace: entryCSN
entryCSN: 20160830072001Z#000001#00#000000
-
replace: modifiersName
modifiersName: cn=[REDACTED], dc=[REDACTED], dc=com, dc=cn
-
replace: modifyTimestamp
modifyTimestamp: 20160830072001Z
-
```



演变的继续：安全考虑

- 加密（原生支持：客户端与服务端的通讯）
 - 证书制作过程建议使用泛域名；
 - 服务器端启用（port：389 -> 636）
 - 客户端启用

```
TLSCipherSuite HIGH:MEDIUM:+SSLv2
TLSCACertificateFile /etc/openssl/cacerts/cacert.pem
TLSCertificateFile /etc/openssl/cacerts/server.cert
TLSCertificateKeyFile /etc/openssl/cacerts/server.key

TLSVerifyClient never
ssl start_tls
```



演变的继续：安全考虑

- 防暴力破解（OpenLDAP原生不支持）
 - 通过PAM模块
 - /etc/pam.d/sshd,当登录失败次数达到5次之后，对该用户进行Lock操作，时间为600秒
 - auth required pam_tally.so onerr=fail deny=5
unlock_time=600



再次继续：服务解耦

- 剥离OpenLDAP服务故障对于平台的影响
 - 通过 nss_initgroups_ignoreusers 模块的定义；
 - # /etc/ldap.conf
 - nss_initgroups_ignoreusers
root,ldap,named,avahi,haldaemon,tomcat,lbs,nagios,nginx,yanfa_
ro,zjxl_ops,oracle,sshd



需求推动演变：对个人账号基于服务器的限制

- 目的：项目组内用户管理
- 涉及到的模块：
 - 模块：ldapns.schema；
 - 配置：Host：HostName；
 - 配置：pam_check_host_attr 和 pam_filter、
nss_base_<map>



需求推动演变：对个人账号基于服务器的限制

配置 `/etc/ldap.conf` 文件，设置 `pam_check_host_attr yes`，参见下例：

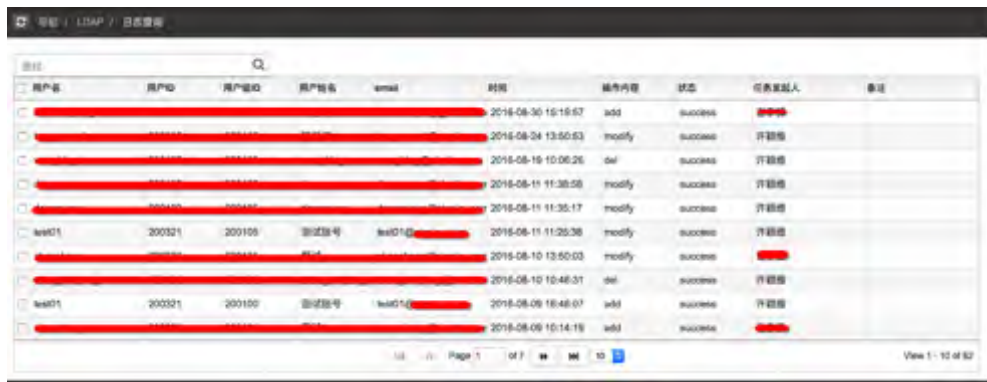
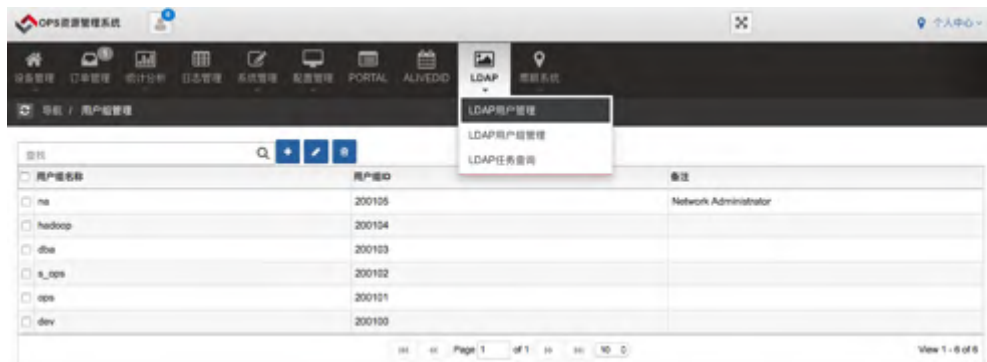
```
# cat /etc/ldap.conf
host 10.65.6.232
base dc=example,dc=com
ssl no
pam_password md5
pam_check_host_attr yes
```

下面是两个用户不同主机属性值的例子：

```
# ldapsearch -x -LLL uid=testat1
dn: uid=testat1,ou=People,dc=example,dc=com
uid: testat1
cn: testat1
objectClass: account
objectClass: posixAccount
objectClass: top
loginShell: /bin/bash
uidNumber: 27651
gidNumber: 27651
homeDirectory: /home/testat1
host: host214.test.example.com
userPassword:
e1NTSEF9dy9ITWJNSVRCOE9pdIA2THdIRDF4UU9hRm9rZWVXL2c=
```



内部需求：高效管理



LDAP用户添加

LDAP帐号*

LDAP密码*

请再次输入密码*

用户组*

请选则用户组

真实姓名*

email*

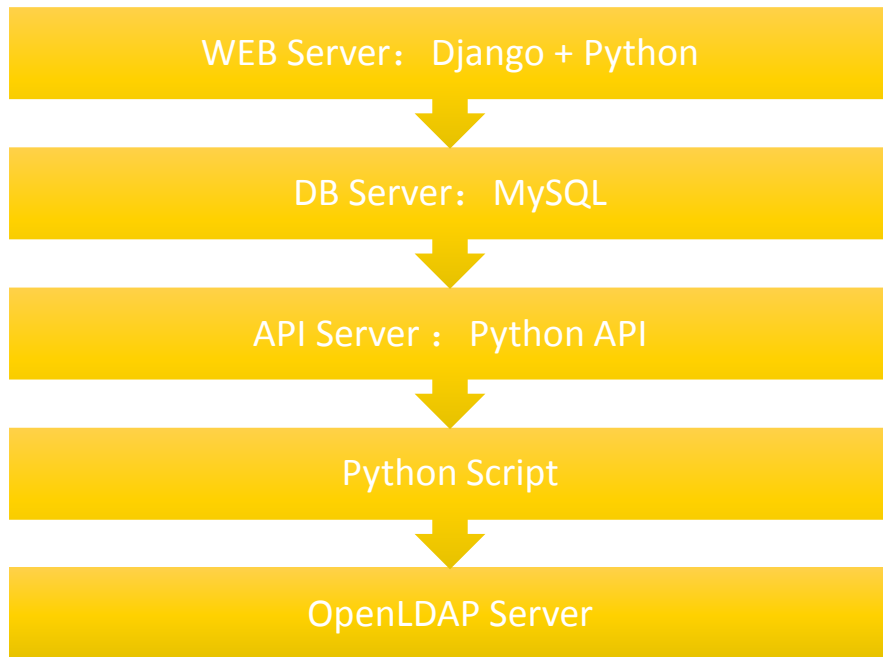
备注

确认保存

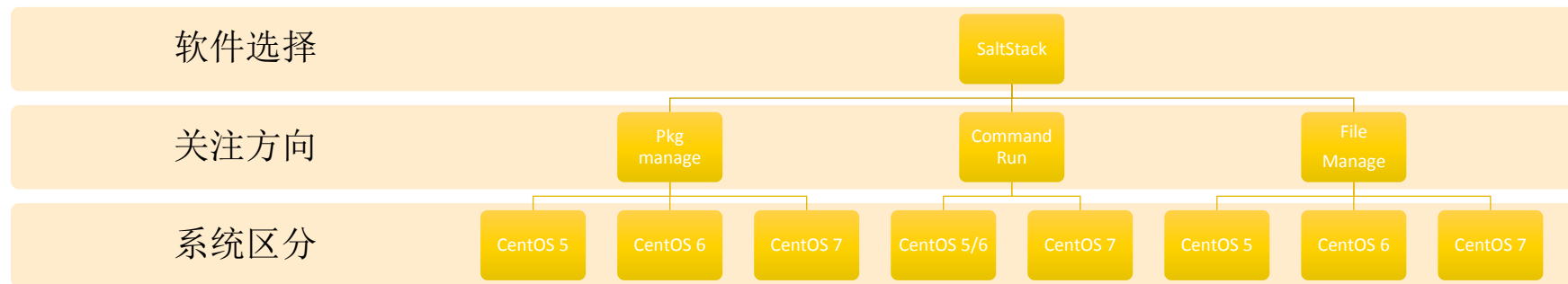


内部需求：高效管理

用户将需求通过WEB页面持久化到MySQL内，PythonScript通过API获取到需求，并生效到OpenLDAP Server内部，并将操作结果返回给API，并持久化到MySQL内。



内部需求：高效安装



NOTE :

SERVICE : nscd or nslcd

PKG: nss_ldap or nss-pam-ldapd

FILE: /etc/sudo-ldap.conf or /etc/sudoers

.....



总结

- **1. 账号管理方面**

- 账号信息管理：包括用户ID、用户账号、用户密码、OU账号（OpenLDAP以OU命名，区分角色组或者部门）、OU_ID等；
- 权限信息管理：根据用户权限需求不同，定义每个归属的OU对应sudoer权限内容，并作为默认权限。而各自的特殊权限只能在相应的服务器上的sudoers里面进行定义；



总结

- **2. 方案架构方面**

- 认证到服务器端进行验证；
- 主从同步实现：一主多从的架构，实现多IDC之间的数据同步；
- 服务业务解耦：对业务归属账号进行忽略定义，避免对业务的影响，及加大OpenLDAP服务的压力；



总结

- **3. 安全管理方面**

- **交互通讯加密**：使用openssl对整个身份认证过程进行加密，保障内部通讯的安全；
- **登录主机限制**：实现对用户拥有登录的设备进行限制，便于管理需要；
- **防止暴力破解**：定义密码失败次数限制，及限制动作的定义；
- **提权功能**：用户必须通过个人账号登录服务器，并使用sudo才能进行提权，方便进行监控与管理；



总结

- **4. 高效管理**

- WEB页面管理：简化操作难度，增加工作效率；
- Salt安装部署：简化操作难度，增加工作效率；



DevOpsDays 即将首次登陆中国



DevOps 之父 Patrick Debois 与您相约

DevOpsDays 北京站 2017年3月18日



门票早鸟价仅限前100名，请从速哟

<http://2017-beijing.devopsdayschina.org/>



GOPS2016
Beijing



想第一时间看到
高效运维社区公众号
的好文章吗？



请打开高效运维社区公众号，点击右上角小人，如右侧所示设置就好





Thanks

高效运维社区
开放运维联盟

荣誉出品

