基于OpenLDAP的统一认证系统

Step1. Install DB6+LDAP

==========

#yum install vim curl wget bind-utils bind-utils vim-enhanced lrzsz gcc make man unzip openssl openssl-devel gnutls cyrus-sasl libedit-devel make autoconf automake curl curl-devel gcc gcc-c++ openssl openssl-devel patch perl cpp glibc libstdc++devel bison

cd /opt/local/src
tar zxvf heimdal-1.5.3.tar.gz
cd heimdal-1.5.3

./configure --without-ipv6 --with-openldap
make
make install
cd ..
tar zxvf db-6.0.30.tar.gz
cd build_unix/
../dist/configure --prefix=/opt/local/BerkeleyDB
make
make install
cd ..
tar zxvf openldap-2.4.39.tar.gz
cd openldap-2.4.39

vim /etc/ld.so.conf
include ld.so.conf.d/*.conf
/opt/local/BerkeleyDB/lib
ldconfig

./configure --prefix=/opt/local/openldap --with-tls=openssl --enable-bdb CPPFLAGS="-I/opt/local/BerkeleyDB/include" LDFLAGS="-L/opt/local/BerkeleyDB/lib"
make depend
make
make install

————————————————————————————————————

Step 2. Config openldap

vim /opt/local/openldap/etc/openldap/slapd.conf
include	/opt/local/openldap/etc/openldap/schema/core.schema
include	/opt/local/openldap/etc/openldap/schema/collective.schema
include	/opt/local/openldap/etc/openldap/schema/corba.schema
include	/opt/local/openldap/etc/openldap/schema/cosine.schema
include	/opt/local/openldap/etc/openldap/schema/duaconf.schema
include	/opt/local/openldap/etc/openldap/schema/dyngroup.schema
include	/opt/local/openldap/etc/openldap/schema/inetorgperson.schema
include	/opt/local/openldap/etc/openldap/schema/java.schema
include	/opt/local/openldap/etc/openldap/schema/misc.schema
include	/opt/local/openldap/etc/openldap/schema/nis.schema
include	/opt/local/openldap/etc/openldap/schema/openldap.schema
include	/opt/local/openldap/etc/openldap/schema/pmi.schema
include	/opt/local/openldap/etc/openldap/schema/ppolicy.schema

access to attr=shadowLastChange,userPassword
	by self write
	by * auth

access to *
	by * read

database	bdb
suffix	"dc=rocnic,dc=com"
rootdn	"cn=Manager,dc=rocnic,dc=com"

rootpw	{SSHA}iPCNnvfzXKaNbxJrtxOoDtXVavMPUxXF

vim root.ldif
dn:dc=cmcctest,dc=com
objectclass: top
objectClass: dcObject
objectClass: organizationalUnit
dc: cmcctest
ou:People

vim /etc/profile
export PATH=$PATH:/opt/local/openldap/bin
source /etc/profile

cp /opt/local/openldap/var/openldap-data/DB_CONFIG.example /opt/local/openldap/var/openldap-data/DB_CONFIG
/opt/local/openldap/libexec/slapd -d 1
vim /etc/rc.d/init.d/slapd

chmod +x /etc/rc.d/init.d/slapd
service slapd start
ldapadd -x -D "cn=Manager,dc=cmcctest,dc=com" -W -f root.ldif

==========

Step 3. Install phpldapadmin + self-service-password

#yum install php-mcrypt.x86_64 php-imap php-ldap //为php增加mcrypt、imap、ldap扩展
service httpd restart
应用程序的目录在/opt/local/ucenter

==========

Test Application Authentication with Openldap

Application1. Client Authentication(Linux System)

CentOS客户端配置：
1.安装openldap客户端：yum install openldap-clients
更改openldap客户端配置文件：
TLS_CACERTDIR /etc/openldap/cacerts
URI ldap://172.16.12.128 //这里写openldap服务器的地址
BASE dc=rocnic,dc=com //查询的basedn

2.配置/etc/nsswitch名称转换服务
该文件由glibc生成，CentOS中默认安装。用于名称转换服务，通常LINUX系统身份验证读取本地文件，要使身份验证查询台欧诺个过LDAP服务器，必须在该文件中找到passwd;shadow;group;三行在file后空格添加"ldap"
passwd: file ldap
shadow: file ldap
group: file ldap

3.配置/etc/sysconfig/authconfig
该文件提供身份验证支持LDAP功能，由authconfig包生成，系统默认安装，配置该文件用来跟踪LDAP身份验证机制是否正确启用。找到以下行，确认值是否正确：
USESYSNETAUTH=yes USESHADOW=yes
USELOCAUTHORIZE=yes
USELDAP=yes
USELDAPAUTH=yes
USEMKHOMEDIR=yes
PASSWDALGORITHM=md5

4.配置/etc/pam.d/system-auth
身份验证服务是实际指向LDAP验证用户身份的服务。可插入身份验证模块(PAM)提供了本地LINUX身份验证服务。pam_unix.so模块是通用模块，使PAM机制对本地的/etc/passwd文件检查用户账号，pam_ldap模块可以用来将身份验证重定向到LDAP目录服务上，身份验证本身由PAM程序执行，它从身份验证候选机制中获取用户名，将其绑定到openldap服务器上，如果绑定成功，PAM会报告说这个用户已经成功通过了q
session required pam_mkhomedir.so skel=/etc/skel/ umask=0022 session optional pam_ldap.so
session optional pam_ldap.so

yum install nss-pam-ldapd
authconfig-tui //GUI Linux Authentication Configuration tools
---

==========

Application2. Client Authentication(USVN)

---

vim /var/www/usvn/config/config.ini
[general]
url.base = "/usvn"
translation.locale = "zh_CN"
timezone = "Asia/Shanghai"
system.locale = "aa_DJ.utf8"
template.name = "usvn"
site.title = "USVN"
site.ico = "medias/usvn/images/logo_small.tiff"
site.logo = "medias/usvn/images/logo_trans.png"
subversion.path = "/var/www/usvn/files/"
subversion.passwd = "/var/www/usvn/files/htpasswd"
subversion.authz = "/var/www/usvn/files/authz"
subversion.url = "http://192.168.107.249/usvn/svn/"
database.adapterName = "PDO_MYSQL"
database.prefix = "usvn_"
database.options.host = "localhost"
database.options.username = "root"
database.options.password = "xxxxxx"
database.options.dbname = "usvn"

```
update.checkforupdate = "0"
update.lastcheckforupdate = "0"
version = "1.0.6"
alwaysUseDatabaseForLogin = "admin"
authAdapterMethod = "ldap"
ldap.options.useStartTls = "0"
ldap.options.useSsl = "0"
ldap.options.bindRequiresDn = "1"
ldap.options.accountCanonicalForm = "0"
ldap.options.allowEmptyPassword = "1"
ldap.options.optReferrals = "0"
ldap.options.host = "192.168.107.64"
ldap.options.port = "389"
ldap.options.baseDn = "dc=cmcctest,dc=com"
ldap.options.username = "cn=Manager,dc=cmcctest,dc=com"
ldap.options.password = "xxxxxx"
ldap.createGroupForUserInDB = "0"
ldap.createUserInDBOnLogin = "1"
```

注意：USVN用的是ssha/md5加密方式，如果加密方式不正确，则认证无法通过。
===
Application3. Client Authentication(Nagios)
——————————————————————
nagios多用户ldap验证配置

```
[root@cacti01 nagios]# cat cgi.cfg l grep -v "^#" lgrep auth
use_authentication=1
use_ssl_authentication=0
authorized_for_system_information=nagiosadmin
authorized_for_configuration_information=nagiosadmin
authorized_for_system_commands=nagiosadmin
authorized_for_all_hosts=*
authorized_for_all_hosts=*
authorized_for_all_service_commands=nagiosadmin
authorized_for_all_host_commands=nagiosadmin
authorized_for_read_only=*
lock_author_names=1
```

```
[root@cacti01 conf.d]# cat nagios.conf
# SAMPLE CONFIG SNIPPETS FOR APACHE WEB SERVER
# Last Modified: 11-26-2005
#
# This file contains examples of entries that need
# to be incorporated into your Apache web server
# configuration file.  Customize the paths, etc. as
# needed to fit your system.

ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin"

<Directory "/usr/local/nagios/sbin">
#  SSLRequireSSL
  Options ExecCGI
  AllowOverride None
  Order allow,deny
  Allow from all
#  Order deny,allow
#  Deny from all
#  Allow from 127.0.0.1
AuthName "Nagios Access"
AuthType Basic
AuthBasicProvider ldap
AuthzLDAPAuthoritative off
AuthLDAPURL ldap://192.168.107.64:389/ou=People,dc=cmcctest,dc=com?uid
Require valid-user
</Directory>

Alias /nagios "/usr/local/nagios/share"

<Directory "/usr/local/nagios/share">
```

```
#  SSLRequireSSL
Options None
AllowOverride None
Order allow,deny
Allow from all
#  Order deny,allow
#  Deny from all
#  Allow from 127.0.0.1
AuthName "Nagios Access"
AuthType Basic
AuthBasicProvider ldap
AuthzLDAPAuthoritative off
AuthLDAPURL ldap://192.168.107.64:389/ou=People,dc=cmcctest,dc=com?uid
Require valid-user
</Directory>
```

==========

Application4. Client Authentication(Zabbix)

————————————————————

Administration-Authentication
Default authentication:LDAP
LDAP host: ldap://192.168.107.64
port: 389
Base DN: ou=people,dc=cmcctest,dc=com
Search attribute:uid
Bind DN: uid=wangshaoqian,ou=People,dc=cmcctest,dc=com
Bind password:xxxxxxx
Test Authentication
Login:Admin　//这个用户必须在zabbix系统中存在
Userpassword:
注意：1.php要安装ldap扩展：yum install php-ldap
　　2.需要用ldap认证的用户必须在zabbix中存在

————————————————————

官方文档部分：
Overview
In Administration → Authentication the user authentication method to Zabbix can be changed. The available methods are internal, LDAP and HTTP authentication.
By default, internal Zabbix authentication is used. To change, click on the button with the method name and press Save.

Internal
Internal Zabbix authentication is used.

LDAP
External LDAP authentication can be used to check user names and passwords. Note that a user must exist in Zabbix as well, however its Zabbix password will not be used.
Zabbix LDAP authentication works at least with Microsoft Active Directory and OpenLDAP.

Configuration parameters:

Parameter	Description
LDAP host	Name of LDAP server. For example: ldap://ldap.zabbix.com For secure LDAP server use ldaps protocol. ldaps://ldap.zabbix.com
Port	Port of LDAP server. Default is 389. For secure LDAP connection port number is normally 636.
Base DN	Base path to search accounts: ou=Users,ou=system (for OpenLDAP), DC=company,DC=com (for Microsoft Active Directory)
Search attribute	LDAP account attribute used for search: uid (for OpenLDAP), sAMAccountName (for Microsoft Active Directory)
Bind DN	LDAP account for binding and searching over the LDAP server, examples: uid=ldap_search,ou=system (for OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (for Microsoft Active Directory) Required, anonymous binding is not supported.
Bind password	LDAP password of the account for binding and searching over the LDAP server.
Test authentication	Header of a section for testing
Login	Name of a test user (which is currently logged in the Zabbix frontend). This user name must exist in the LDAP server. Zabbix will not activate LDAP authentication if it is unable to authenticate the test user.
user password	LDAP password of the test user

It is recommended to create a separate LDAP account (Bind DN) to perform binding and searching over the LDAP server with minimal privileges in the LDAP instead of using real user accounts (used for logging in the Zabbix frontend).
Such an approach provides more security and does not require changing the Bind password when the user changes his own password in the LDAP server.
In the table above it's ldap_search account name.

Some user groups can still be authorized by Zabbix. These groups must have frontend access set to Internal.

HTTP

Apache-based (HTTP) authentication can be used to check user names and passwords. Note that a user must exist in Zabbix as well, however its Zabbix password will not be used.

Warning: Be careful! Make sure that Apache authentication is configured and works properly before switching it on.

note:In case of Apache authentication all users (even with frontend access set to Internal) will be authorized by Apache, not by Zabbix!