

# Manage password policy in OpenLDAP

Clément OUDOT  
[coudot@linagora.com](mailto:coudot@linagora.com)



First time you see me? Let's introduce:



LDAPcoholic since many years  
Fake developer, real hacker





Let's begin with the password  
policy draft (Behera draft)





Well, not really.  
The first draft (version 0)  
was written in 1999.

A draft? Is it  
not a standard?



The latest version (version 10) was published in 2009



This draft is expired  
since February 2010





Of course!  
Most of LDAP servers  
implement it.

So, can we use it?





What are you waiting for? Explain me how it works!







Ok, let me do the LDAP client. You will play the LDAP server.







Ok, I send you an BIND operation with the extended control 1.3.6.1.4.1.42.2.27.8.5.1



I see your password is expired, I refuse the BIND and I send a flag in the response control.



Thanks to this response control, I can advertise the user.



See, it's easy! Client and server just need to know how to manage the control.





With which LDAP operations  
can we use this control?

BIND for authentication.  
MOD and PASSMOD for  
password change.





For authentication, it defines  
account locking, password  
expiration and password reset



For modification, it can check password size,  
presence in history, password quality.




Niark Niark

With this, administrators  
will have the power to  
bother all their users...





Let me now present you  
my friend openLDAP



Hi! I am the fastest  
LDAP server on earth!



I own a password policy overlay since many years



I support version 9 of the Behera draft and let the possibility to implement a custom password checker module





I imagine that configuring password policy overlay is a nightmare!

Calm down, you just need a brain!





First, load the overlay: `olcModuleLoad: ppolicy.1a`

Then configure it:

```
dn: olcOverlay={1}ppolicy,olcDatabase={1}bdb,cn=config
objectClass: olcOverlayConfig
objectClass: olcPPolicyConfig
olcOverlay: {1}ppolicy
olcPPolicyDefault: ou=default,ou=ppolicy,dc=example,dc=com
olcPPolicyHashCleartext: TRUE
olcPPolicyUseLockout: FALSE
olcPPolicyForwardUpdates: FALSE
```





So is it over? That was easy!

No, we now need to configure the policy



Policy configuration is an entry in the LDAP directory

The first lines of the entry are:

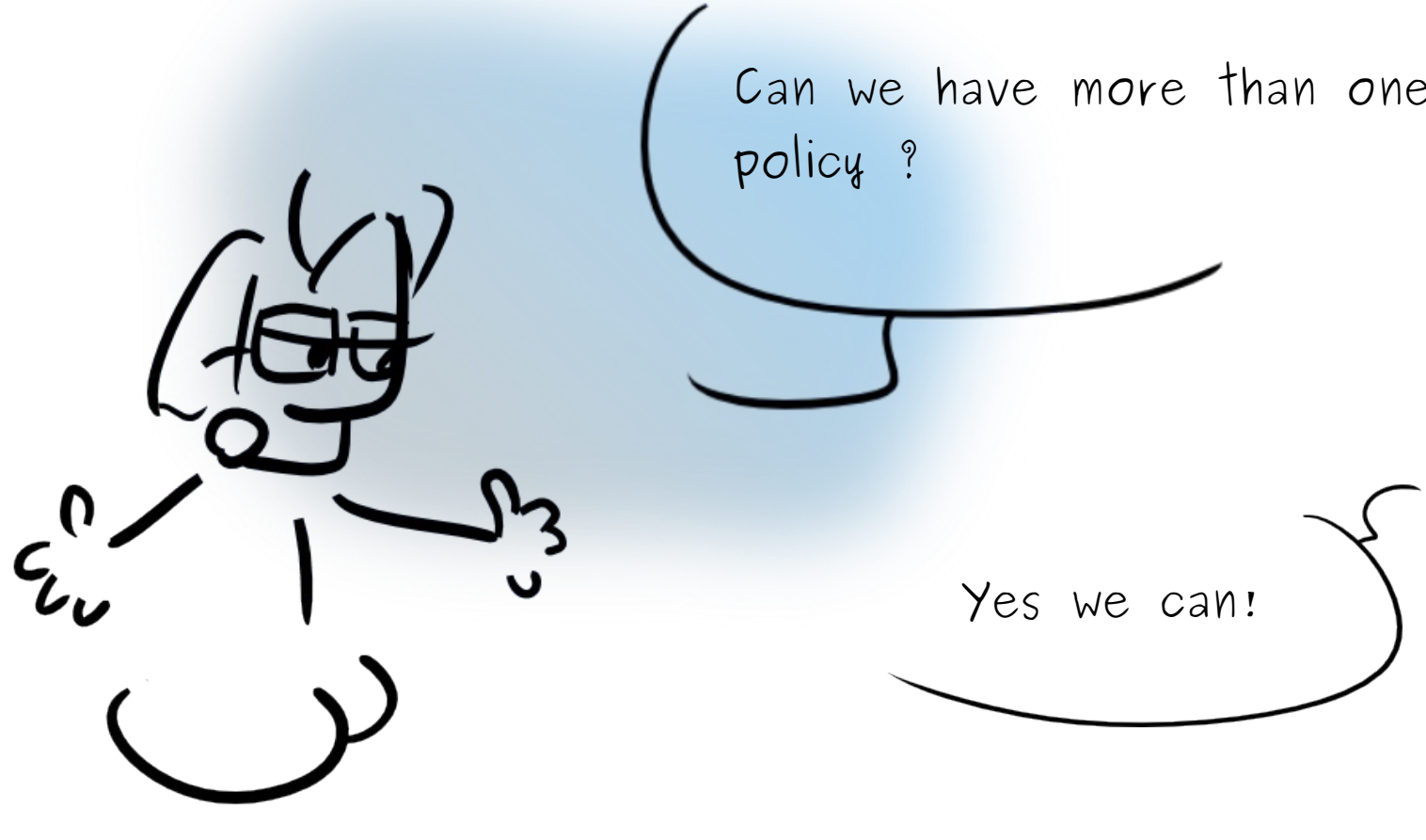
```
dn: ou=default,ou=ppolicy,dc=example,dc=com
objectClass: pwdPolicy
objectClass: pwdPolicyChecker
objectClass: organizationalUnit
objectClass: top
ou: default
```



```
pwdAllowUserChange: TRUE
pwdAttribute: userPassword
pwdCheckModule: check_password.so
pwdCheckQuality: 2
pwdExpireWarning: 0
pwdInHistory: 10
pwdLockout: TRUE
pwdMaxAge: 31536000
pwdMinAge: 600
pwdMaxFailure: 10
pwdMinLength: 8
pwdMustChange: TRUE
PwdsafeModify : FALSE
```

Then all parameters are  
attributes of this entry





Can we have more than one policy ?



Yes we can!



Just create another policy configuration entry

Then link it to a user account:


```
dn: uid=bobama,ou=users,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
ObjectClass : person
objectClass: top
uid : bobama
cn : Barack OBAMA
sn : OBAMA
userPassword: michellemabelle
pwdPolicySubentry : ou=nsa,ou=ppolicy,dc=example,dc=com
```





Did you heard about LDAP  
Tool Box project?

Yes, they provide a  
password checker module  
and OpenLDAP package  
for Debian and Centos





They also package some  
contributed overlays like  
lastbind and smb5pwd

Indeed,  
good job!







This is all folks!  
Any question?





Saisissez votre bulle



#### INSTRUCTIONS

**Flèche bas:** rétrécir  
**Flèche haut:** agrandir  
**Gauche/Droite:** retournement horizontal  
**Del:** effacer l'objet

ENREGISTRER VOTRE IMAGE

REDIMENSIONNER LA FENÊTRE  
800x600 | 640x480 | 320x240



Made with Gégé  
<https://framalab.org/gknd-creator/>

Credits to Simon "Gee" Giraudot  
Creative Commons By-Sa

