

#####.
#####/
#####_#####_
#####.

#####

#####,#####
#####,#####
#####.

#####-
#####-
#####,
#####.
#####-
#####.

2.

#.500#####.#####
#####.#####
#####.

#####;#####
#####,

#####/#####24#####.#####
#####/#####24#####.

#####.
#####,

#####.

2.1.



####

#####-
#####-
####.

2.1.1.

#####:

#####

```
TLSCertificateFile -/path/to/your/cert.crt
TLSCertificateKeyFile -/path/to/your/cert.key
TLSCACertificateFile -/path/to/your/cacert.crt
```

####, ssf=128 ##### ##### ## ##### 128#### ##### ### ## #####,
#####. ##### ##### ## ##### ##### ## ## #####
##, ## ##### ## ##### ##, ## ##### #####
#####.

cert.crt, cert.key, ### cacert.crt ##### ## ##### ## ##### you
#####. ## ## ##### # ##### ##, ## ## ##
##:

2. ##### ##

```
% openssl genrsa --out cert.key 1024
Generating RSA private key, 1024 bit long modulus
.....++++++
...++++++
e is 65537 (0x10001)
% openssl req --new --key cert.key --out cert.csr
```

#####. ## ## #####
##; #####, ## ## ##### ## ##### ## ##
#####. ## ## ##, ## ## #####
server.example.org. ##### ## ## ##
#####. ##### ## ## ##, ##
##.

#####, ## ##### ## ## ##:

3.

```
% openssl x509 --req --in cert.csr --days 365 --signkey cert.key --out cert.crt
Signature ok
subject=/C=AU/ST=Some-State/O=Internet Widgits Pty Ltd
Getting Private key
```


#####

```
##### ldapsearch -Z ##### -Z #####  
#####. #####, ##### #####; #####  
##### #####. ##### #####(1)##### s_client ##### s_server #####  
##### #####.
```

2.2. ##### ## ###

```
##### ## ##### ##### ##### ##### #####-
##### ## ##### ##### #####. ##### ## #####
##### ## ##### ##### #####. ##### ## ##
##### uid ##### ## ##### ##### ##### userPassword #####
#####, ##### ## #####.
```


#####.

```
#### ##### ##### ## dc=example,dc=org. #### ##### ##### #####
##### ##### ##### ## ##### ## ##### ##### ou=people,base, ## #####
##### ## #####. ##### ##### ## ##### ##### ## #####.
```

people ##### ## ## ## ## ##:

```
dn: ou=people,dc=example,dc=org
objectClass: top
objectClass: organizationalUnit
ou: people
```

#####.

```
#####  
##### people,#####  
#####.  
#####  
##### inetOrgPerson,  
#####.  
#####  
##### slapd.conf.
```

```
### ##### ##### ## ##### ### ## person ##### #####. ## ### ## ##### inetOrgPerson,
### ##### ### #####, ##### ##### ## sn ##### ## #####.
```

```
## ### # ##### testuser, ### ##### ##### ##:
```

```
dn: uid=tuser,ou=people,dc=example,dc=org
objectClass: person
objectClass: posixAccount
objectClass: shadowAccount
objectClass: top
uidNumber: 10000
gidNumber: 10000
```

#####

```
homeDirectory: /home/tuser
loginShell: /bin/csh
uid: tuser
cn: tuser
```

```
# ##### ## ##### ##### ## 10000 ## ##### ##### ##### #####;
### ## ##### ##### ##### ## ##### ## ##### ## 65536.

## ##### ##### #####. ##### ## ##### ## ##### ## ## ##
### ## ##### #####:
```

```
dn: ou=groups,dc=example,dc=org
objectClass: top
objectClass: organizationalUnit
ou: groups

dn: cn=tuser,ou=groups,dc=example,dc=org
objectClass: posixGroup
objectClass: top
gidNumber: 10000
cn: tuser
```

```
## ##### ##### #####, ### ## ## slapadd ## ldapadd ## # #####
#####. #####, ### ## ## #####/#####.

### ldapsearch ##### ## ## ##### ##### ## #####. ## ##
####, ##### ## ##### ## ## ## ##
#####.
```

3.

```
### ##### ##### ##### ##### ##### ##### ##### 2.1.3, #####-
##### ## #####, ### ## ## ##### ##### ##### ## ##
#### ## ##### ##/#####24##### ## ##### ## ##.
```

```
##### ##### ## ## ##### ## ##### ##### ## #####-
##, #####/##_ ##### ## ##/##_.
```

3.1.

```
#####/##_ ##### ## ##### ## /usr/local/etc/ldap.conf.
```



####

```
#### ## # different file #### ## #####
##### ##, /usr/local/etc/openldap/ldap.conf; #####, ## #####
```

```
#### ## ### #####; ## #### ## # ##### ## ####
####. ### ## ### ## #### #####, ##### ## ldap.conf ####
#### /usr/local/etc/ldap.conf.
```

```
## ### ##### uid #####. ## ##### ## (##### ##  
## ### #####), ### ## pam_login_attribute ##### ## ldap.conf:
```

pam_login_attribute uid

3.1.1.

```
#####  
(#####  
#####).
```

```
auth sufficient /usr/local/lib/pam_ldap.so no_warn
```

```
#####  
#####.  
#####
```


#####

```
## ##### ## ## ## ## #####. ## ##### #####  
##### ## ##### ##### ## /etc/pam.d ## ##### #####--  
#####.
```

3.2. ####

```
### ## ## ##### ##### ##### ##### ## #####. ##, ### #####, ## # ##### ##  
##### ## ##### 1001, ## ##### ##### ##### ## ## ## ## 1001, ### ## #####  
### bob ## ted ## ##### ## ## ## ##.
```

```
### ##### ## ##### ##### ## ##### ## ## ##, ## ##### ## ##### ## ## #####  
#####.
```

```
### ###/###_##### ##### #####. ## ##### ## ##### ##### ## #####--  
##/###_#####, ### ##### ## ##### ## ##### ##### ## ## #####. ##--  
#####, ##### ## ##### ## ##### ## ##### /etc/nsswitch.conf ## ##### ## ## ##--  
#####. ##### ##### ## ##### #####:
```

```
group: compat  
passwd: compat
```

####

```
group: files ldap  
passwd: files ldap
```

```
#### ##### ## ## ## ##### ## ##### ## ## ##.
```

```
#####! ## ##### ## ##### #####.
```

3.3.

```
#####, ## ## ## ## ## ##### ##### ## ## #####  
##### ##### #####(1). ##### ## ##, #####  
##### # #####. # ##### #####. ##### ## ##  
##### ## #####, ##### ## #####  
## ##### ##, ## ##### 4.3, #####
```

6. #####

```
#!/bin/sh  
stty --echo
```

#####

```
read -p -"Old Password: -" oldp; echo
read -p -"New Password: -" np1; echo
read -p -"Retype New Password: -" np2; echo
stty echo

if [ -"$np1" -!= -"$np2" -]; then
    echo -"Passwords do not match."
    exit 1
fi

ldappasswd -D uid="$USER",ou=people,dc=example,dc=org \
--w -"$oldp" \
--a -"$oldp" \
--s -"$np1"
```



#####

#####, ### ##### ##—
##—
#####. ## ### ## ##### ##### #####, ## ##### ##### ##
security.bsd.see_other_uids #####:#####

```
# sysctl security.bsd.see_other_uids=0.
```

(### #####) ##### ## ## ##### ## ##### # ##—
#####, ## ##### # ### #####. ### ##### ## ##### ## # ##### #####
#####. ## ##### ## ##### ## ##### #####, ### ## ## ##.

7. ##### ##### ##

```
require 'ldap'
require 'base64'
require 'digest'
require 'password' # ruby-password

ldap_server = -"ldap.example.org"
luser = -"uid=#{ENV['USER']},ou=people,dc=example,dc=org"

# get the new password, check it, and create a salted hash from it
def get_password
    pwd1 = Password.get("New Password: -")
    pwd2 = Password.get("Retype New Password: -")
```

#####

```
raise if pwd1 != pwd2
pwd1.check # check password strength

salt = rand.to_s.gsub(/0\./, "-")
pass = pwd1.to_s
hash = -"{SSHA}" + Base64.encode64(Digest::SHA1.digest("#{pass}#{salt}")) + salt).chomp!
return hash
end

oldp = Password.get("Old Password: -")
newp = get_password

# We'll just replace it. That we can bind proves that we either know
# the old password or are an admin.

replace = LDAP::Mod.new(LDAP::LDAP_MOD_REPLACE -| -
  LDAP::LDAP_MOD_BVALUES,
    -"userPassword",
    [newp])

conn = LDAP::SSLConn.new(ldap_server, 389, true)
conn.set_option(LDAP::LDAP_OPT_PROTOCOL_VERSION, 3)
conn.bind(luser, oldp)
conn.modify(luser, [replace])
```


#####, #####) ##### sh #####.

4.

#####, #####
master.passwd #####
#####

#####, #####
#####

4.1.

#####, # ##### uidNumber ##### 0 ### root #####!

#####, userPassword #####

#####, slaped.conf:

#####

8.

```
access to dn.subtree="ou=people,dc=example,dc=org"
  attrs=userPassword
  by self write
  by anonymous auth
  by * none

access to *
  by self write
  by * read
```

userPassword #####, ##### #####
#####.

, ##### ## ##### ##### ##### ## ##### ## #####—
####. ## #####, ##### ## ##### ## ##### (##### ## ##### ## #####
#####), ##### ## uidNumber. ## ##### ## ##, ##### ##
##

9.

```
access to dn.subtree="ou=people,dc=example,dc=org"
  attrs=userPassword
  by self write
  by anonymous auth
  by * none

access to attrs=homeDirectory,uidNumber,gidNumber
  by * read

access to *
  by self write
  by * read
```

#####.

4.2. Root

root ## ##### ##### ## ## ##### ## ## ##### ## ##—
#####. ##### #####, ## #####, ## ## ##, ## ##

#####

4.3.

userPassword ##### ## ## #####
#: ## #####. ##### ## ##### ## ## ##### 64 #####, #####

##.

##, ##, ## ##### ##### ## # ##### ##### ##, ## ##
(#####). ##### ## ##### ##### ##### ## ## ## #####.

#.

#####, ##### ##
#####.

#####/##_##### ## # ## ##### ##### #####; ##### ##
#####. ## ## #####
#####, ## ## ##### ## ## #####
#####.

#####/### ## # [#\(8\)](#)##### ##### ## ## ## ## #####
#####. ## ## ## ## #####, ## ##### ##### ##. ## #####
(### -x ###) ### (#####).

#####/##### ## # ##### ##### ## ##### ## ## ##### ##—
###. ## ##### (## #####) ## ##### ## ##
EDITOR #####. ##### ## ## ##
#####.

#####/##### ## ## ##### ## ##### ## #####
##. ##### ## ##### ## ## ## ##### ## ##
##.

#.

#####, ## #####
#####, ##### ##### ## ##
#####. ##### ## #####, ## ## ##
#####, ## #####
#####.

[#####\(1\)](#) ## ##.

#####, ## #####
##. ## #####

#####

#.1. ##### #

```
% openssl genrsa --out root.key 1024
% openssl req --new --key root.key --out root.csr
% openssl x509 --req --days 1024 --in root.csr --signkey root.key --out root.crt
```


#####

####, #####
ldap-server-one.csr. ##### root.key, ##
ldap-server-one.*



####

#####

-CA -CAkey -signkey:

#.2. ##### #

```
% openssl x509 --req --days 1024 \
-in ldap-server-one.csr --CA root.crt --CAkey root.key \
-out ldap-server-one.crt
```

#####

root.crt (## *certificate*,
#####) ## TLSCACertificateFile #####
ldap.conf.