

**NAME**

smbpasswd - change a user's SMB password

**SYNOPSIS**

**smbpasswd** [-a] [-c <config file>] [-x] [-d] [-e] [-D debuglevel] [-n] [-r <remote machine>] [-R <name resolve order>] [-m] [-U username[%password]] [-h] [-s] [-w pass] [-W] [-i] [-L] [username]

**DESCRIPTION**

This tool is part of the **samba(7)** suite.

The smbpasswd program has several different functions, depending on whether it is run by the **root** user or not. When run as a normal user it allows the user to change the password used for their SMB sessions on any machines that store SMB passwords.

By default (when run with no arguments) it will attempt to change the current user's SMB password on the local machine. This is similar to the way the **passwd(1)** program works.

**smbpasswd** differs from how the passwd program works however in that it is not **setuid root** but works in a client-server mode and communicates with a locally running **smbd(8)**. As a consequence in order for this to succeed the smbd daemon must be running on the local machine. On a UNIX machine the encrypted SMB passwords are usually stored in the **smbpasswd(5)** file.

When run by an ordinary user with no options, smbpasswd will prompt them for their old SMB password and then ask them for their new password twice, to ensure that the new password was typed correctly. No passwords will be echoed on the screen whilst being typed. If you have a blank SMB password (specified by the string "NO PASSWORD" in the smbpasswd file) then just press the <Enter> key when asked for your old password.

smbpasswd can also be used by a normal user to change their SMB password on remote machines, such as Windows NT Primary Domain Controllers. See the (-r) and -U options below.

When run by root, smbpasswd allows new users to be added and deleted in the smbpasswd file, as well as allows changes to the attributes of the user in this file to be made. When run by root,

**smbpasswd** accesses the local smbpasswd file directly, thus enabling changes to be made even if smbd is not running.

**OPTIONS**

- a This option specifies that the username following should be added to the local smbpasswd file, with the new password typed (type <Enter> for the old password). This option is ignored if the username following already exists in the smbpasswd file and it is treated like a regular change password command. Note that the default passwd backends require the user to already exist in the system password file (usually */etc/passwd*), else the request to add the user will fail.

This option is only available when running smbpasswd as root.

- c This option can be used to specify the path and file name of the *smb.conf* configuration file when it is important to use other than the default file and / or location.
- x This option specifies that the username following should be deleted from the local smbpasswd file.

This option is only available when running smbpasswd as root.

- d This option specifies that the username following should be **disabled** in the local smbpasswd file. This is done by writing a '**D**' flag into the account control space in the smbpasswd file. Once this is done all attempts to authenticate via SMB using this username will fail.

If the smbpasswd file is in the 'old' format (pre-Samba 2.0 format) there is no space in the user's password entry to write this information and the command will FAIL. See **smbpasswd(5)** for details on the 'old' and new password file formats.

This option is only available when running smbpasswd as root.

- e This option specifies that the username following should be **enabled** in the local smbpasswd file, if the account was previously disabled. If the account was not disabled this option has no effect. Once the account is enabled then the user will be able to authenticate via SMB once again.

If the smbpasswd file is in the 'old' format, then

**smbpasswd** will FAIL to enable the account. See **smbpasswd(5)** for details on the 'old' and new password file formats.

This option is only available when running smbpasswd as root.

- D debuglevel

*debuglevel* is an integer from 0 to 10. The default value if this parameter is not specified is zero.

The higher this value, the more detail will be logged to the log files about the activities of smbpasswd. At level 0, only critical errors and serious warnings will be logged.

Levels above 1 will generate considerable amounts of log data, and should only be used when investigating a problem. Levels above 3 are designed for use only by developers and generate HUGE amounts of log data, most of which is extremely cryptic.

- n This option specifies that the username following should have their password set to null (i.e. a blank password) in the local smbpasswd file. This is done by writing the string "NO PASSWORD" as the first part of the first password stored in the smbpasswd file.

Note that to allow users to logon to a Samba server once the password has been set to "NO PASSWORD" in the smbpasswd file the administrator must set the following parameter in the [global] section of the *smb.conf* file :

**null passwords = yes**

This option is only available when running smbpasswd as root.

- r remote machine name

This option allows a user to specify what machine they wish to change their password on. Without this parameter smbpasswd defaults to the local host. The *remote machine name* is the NetBIOS name of the SMB/CIFS server to contact to attempt the password change. This name is resolved into an IP address using the standard name resolution mechanism in all programs of the Samba suite. See the *-R name resolve order* parameter for details on changing this resolving mechanism.

The username whose password is changed is that of the current UNIX logged on user. See the *-U username* parameter for details on changing the password for a different username.

Note that if changing a Windows NT Domain password the remote machine specified must be the Primary Domain Controller for the domain (Backup Domain Controllers only have a read-only copy of the user account database and will not allow the password change).

**Note** that Windows 95/98 do not have a real password database so it is not possible to change passwords specifying a Win95/98 machine as remote machine target.

- R name resolve order

This option allows the user of smbpasswd to determine what name resolution services to use when looking up the NetBIOS name of the host being connected to.

The options are : "lmhosts", "host", "wins" and "bcast". They cause names to be resolved as follows:

&#8226;

**lmhosts:** Lookup an IP address in the Samba `lmhosts` file. If the line in `lmhosts` has no name type attached to the NetBIOS name (see the **lmhosts(5)** for details) then any name type matches for lookup.

&#8226;

**host:** Do a standard host name to IP address resolution, using the system `/etc/hosts`, NIS, or DNS lookups. This method of name resolution is operating system depended for instance on IRIX or Solaris this may be controlled by the `/etc/nsswitch.conf` file). Note that this method is only used if the NetBIOS name type being queried is the `0x20` (server) name type, otherwise it is ignored.

&#8226;

**wins:** Query a name with the IP address listed in the `wins server` parameter. If no WINS server has been specified this method will be ignored.

&#8226;

**bcast:** Do a broadcast on each of the known local interfaces listed in the `interfaces` parameter. This is the least reliable of the name resolution methods as it depends on the target host being on a locally connected subnet.

The default order is **lmhosts**, **host**, **wins**, **bcast** and without this parameter or any entry in the **smb.conf(5)** file the name resolution methods will be attempted in this order.

-m

This option tells `smbpasswd` that the account being changed is a MACHINE account. Currently this is used when Samba is being used as an NT Primary Domain Controller.

This option is only available when running `smbpasswd` as root.

-U username

This option may only be used in conjunction with the `-r` option. When changing a password on a remote machine it allows the user to specify the user name on that machine whose password will be changed. It is present to allow users who have different user names on different systems to change these passwords.

-h This option prints the help string for

**smbpasswd**, selecting the correct one for running as root or as an ordinary user.

-s This option causes `smbpasswd` to be silent (i.e. not issue prompts) and to read its old and new passwords from standard input, rather than from `/dev/tty` (like the **passwd(1)** program does). This option is to aid people writing scripts to drive `smbpasswd`

-w password

This parameter is only available if Samba has been compiled with LDAP support. The `-w` switch is used to specify the password to be used with the `ldap admin dn`. Note that the password is stored in the `secrets.tdb` and is keyed off of the admin's DN. This means that if the value of `ldap admin dn` ever changes, the password will need to be manually updated as well.

-W

**NOTE:** This option is same as `-w` except that the password should be entered using `stdin`.

This parameter is only available if Samba has been compiled with LDAP support. The `-W` switch is used to specify the password to be used with the `ldap admin dn`. Note that the password is stored in the `secrets.tdb` and is keyed off of the admin's DN. This means that if the value of `ldap admin dn` ever changes, the password will need to be manually updated as well.

-i This option tells `smbpasswd` that the account being changed is an interdomain trust account. Currently this is used when Samba is being used as an NT Primary Domain Controller. The account contains the info about another trusted domain.

This option is only available when running `smbpasswd` as root.

-L Run in local mode.

username

This specifies the username for all of the **root only** options to operate on. Only root can specify this parameter as only root has the permission needed to modify attributes directly in the local smbpasswd file.

## NOTES

Since **smbpasswd** works in client-server mode communicating with a local smbd for a non-root user then the smbd daemon must be running for this to work. A common problem is to add a restriction to the hosts that may access the

**smbd** running on the local machine by specifying either *allow hosts* or *deny hosts* entry in the **smb.conf**(5) file and neglecting to allow "localhost" access to the smbd.

In addition, the smbpasswd command is only useful if Samba has been set up to use encrypted passwords.

## VERSION

This man page is correct for version 3.0 of the Samba suite.

## SEE ALSO

**smbpasswd**(5), **Samba**(7).

## AUTHOR

The original Samba software and related utilities were created by Andrew Tridgell. Samba is now developed by the Samba Team as an Open Source project similar to the way the Linux kernel is developed.

The original Samba man pages were written by Karl Auer. The man page sources were converted to YODL format (another excellent piece of Open Source software, available at <ftp://ftp.icce.rug.nl/pub/unix/>) and updated for the Samba 2.0 release by Jeremy Allison. The conversion to DocBook for Samba 2.2 was done by Gerald Carter. The conversion to DocBook XML 4.2 for Samba 3.0 was done by Alexander Bokovoy.