NAME

login.defs - shadow password suite configuration

DESCRIPTION

The /etc/login.defs file defines the site-specific configuration for the shadow password suite. This file is required. Absence of this file will not prevent system operation, but will probably result in undesirable operation.

This file is a readable text file, each line of the file describing one configuration parameter. The lines consist of a configuration name and value, separated by whitespace. Blank lines and comment lines are ignored. Comments are introduced with a # pound sign and the pound sign must be the first non-white character of the line.

Parameter values may be of four types: strings, booleans, numbers, and long numbers. A string is comprised of any printable characters. A boolean should be either the value *yes* or *no*. An undefined boolean parameter or one with a value other than these will be given a *no* value. Numbers (both regular and long) may be either decimal values, octal values (precede the value with *0*) or hexadecimal values (precede the value with *0x*). The maximum value of the regular and long numeric parameters is machine-dependent.

The following configuration items are provided:

CHFN_RESTRICT (string)

This parameter specifies which values in the *gecos* field of the /etc/passwd file may be changed by regular users using the **chfn** program. It can be any combination of letters *f*, *r*, *w*, *h*, for Full name, Room number, Work phone, and Home phone, respectively. For backward compatibility, *yes* is equivalent to *rwh* and *no* is equivalent to *frwh*. If not specified, only the superuser can make any changes. The most restrictive setting is better achieved by not installing **chfn** SUID.

CONSOLE_GROUPS (string)

List of groups to add to the users supplementary groups set when logging in on the console (as determined by the CONSOLE setting). Default is none.

Use with caution - it is possible for users to gain permanent access to these groups, even when not logged in on the console.

CREATE_HOME (boolean)

Indicate if a home directory should be created by default for new users.

This setting does not apply to system users, and can be overridden on the command line.

DEFAULT_HOME (boolean)

Indicate if login is allowed if we cant cd to the home directory. Default is no.

If set to *yes*, the user will login in the root (/) directory if it is not possible to cd to her home directory.

ENCRYPT_METHOD (string)

This defines the system default encryption algorithm for encrypting passwords (if no algorithm are specified on the command line).

It can take one of these values: *DES* (default), *MD5*, *SHA256*, *SHA512*.

Note: this parameter overrides the **MD5_CRYPT_ENAB** variable.

Note: This only affect the generation of group passwords. The generation of user passwords is done by PAM and subject to the PAM configuration. It is recommended to set this variable consistently with the PAM configuration.

ENV_HZ (string)

If set, it will be used to define the HZ environment variable when a user login. The value must be preceded by *HZ=*. A common value on Linux is *HZ=100*.

The **HZ** environment variable is only set when the user (the superuser) logs in with **sulogin**.

ENV_PATH (string)
> If set, it will be used to define the PATH environment variable when a regular user login. The value is a colon separated list of paths (for example */bin:/usr/bin*) and can be preceded by *PATH=*. The default value is *PATH=/bin:/usr/bin.*

ENV_SUPATH (string)
> If set, it will be used to define the PATH environment variable when the superuser login. The value is a colon separated list of paths (for example */sbin:/bin:/usr/sbin:/usr/bin*) and can be preceded by *PATH=*. The default value is *PATH=/sbin:/bin:/usr/sbin:/usr/bin.*

ERASECHAR (number)
> Terminal ERASE character (*010* = backspace, *0177* = DEL).
>
> The value can be prefixed 0 for an octal value, or 0x for an hexadecimal value.

FAIL_DELAY (number)
> Delay in seconds before being allowed another attempt after a login failure.

FAKE_SHELL (string)
> If set, **login** will execute this shell instead of the users shell specified in /etc/passwd.

GID_MAX (number), **GID_MIN** (number)
> Range of group IDs used for the creation of regular groups by **useradd**, **groupadd**, or **newusers**.
>
> The default value for **GID_MIN** (resp. **GID_MAX**) is 1000 (resp. 60000).

HUSHLOGIN_FILE (string)
> If defined, this file can inhibit all the usual chatter during the login sequence. If a full pathname is specified, then hushed mode will be enabled if the users name or shell are found in the file. If not a full pathname, then hushed mode will be enabled if the file exists in the users home directory.

KILLCHAR (number)
> Terminal KILL character (*025* = CTRL/U).
>
> The value can be prefixed 0 for an octal value, or 0x for an hexadecimal value.

LOG_OK_LOGINS (boolean)
> Enable logging of successful logins.

LOG_UNKFAIL_ENAB (boolean)
> Enable display of unknown usernames when login failures are recorded.
>
> Note: logging unknown usernames may be a security issue if an user enter her password instead of her login name.

LOGIN_RETRIES (number)
> Maximum number of login retries in case of bad password.
>
> This will most likely be overridden by PAM, since the default pam_unix module has its own built in of 3 retries. However, this is a safe fallback in case you are using an authentication module that does not enforce PAM_MAXTRIES.

LOGIN_TIMEOUT (number)
> Max time in seconds for login.

MAIL_DIR (string)
> The mail spool directory. This is needed to manipulate the mailbox when its corresponding user account is modified or deleted. If not specified, a compile-time default is used.

MAIL_FILE (string)
> Defines the location of the users mail spool files relatively to their home directory.

The **MAIL_DIR** and **MAIL_FILE** variables are used by **useradd**, **usermod**, and **userdel** to create, move, or delete the users mail spool.

MAX_MEMBERS_PER_GROUP (number)
> Maximum members per group entry. When the maximum is reached, a new group entry (line) is started in /etc/group (with the same name, same password, and same GID).
>
> The default value is 0, meaning that there are no limits in the number of members in a group.
>
> This feature (split group) permits to limit the length of lines in the group file. This is useful to make sure that lines for NIS groups are not larger than 1024 characters.
>
> If you need to enforce such limit, you can use 25.
>
> Note: split groups may not be supported by all tools (even in the Shadow toolsuite). You should not use this variable unless you really need it.

MD5_CRYPT_ENAB (boolean)
> Indicate if passwords must be encrypted using the MD5-based algorithm. If set to *yes*, new passwords will be encrypted using the MD5-based algorithm compatible with the one used by recent releases of FreeBSD. It supports passwords of unlimited length and longer salt strings. Set to *no* if you need to copy encrypted passwords to other systems which dont understand the new algorithm. Default is *no*.
>
> This variable is superseded by the **ENCRYPT_METHOD** variable or by any command line option used to configure the encryption algorithm.
>
> This variable is deprecated. You should use **ENCRYPT_METHOD**.
>
> Note: This only affect the generation of group passwords. The generation of user passwords is done by PAM and subject to the PAM configuration. It is recommended to set this variable consistently with the PAM configuration.

PASS_MAX_DAYS (number)
> The maximum number of days a password may be used. If the password is older than this, a password change will be forced. If not specified, -1 will be assumed (which disables the restriction).

PASS_MIN_DAYS (number)
> The minimum number of days allowed between password changes. Any password changes attempted sooner than this will be rejected. If not specified, -1 will be assumed (which disables the restriction).

PASS_WARN_AGE (number)
> The number of days warning given before a password expires. A zero means warning is given only upon the day of expiration, a negative value means no warning is given. If not specified, no warning will be provided.

PASS_MAX_DAYS, **PASS_MIN_DAYS** and **PASS_WARN_AGE** are only used at the time of account creation. Any changes to these settings wont affect existing accounts.

SHA_CRYPT_MIN_ROUNDS (number), **SHA_CRYPT_MAX_ROUNDS** (number)
> When **ENCRYPT_METHOD** is set to *SHA256* or *SHA512*, this defines the number of SHA rounds used by the encryption algorithm by default (when the number of rounds is not specified on the command line).
>
> With a lot of rounds, it is more difficult to brute forcing the password. But note also that more CPU resources will be needed to authenticate users.
>
> If not specified, the libc will choose the default number of rounds (5000).
>
> The values must be inside the 1000-999,999,999 range.

If only one of the **SHA_CRYPT_MIN_ROUNDS** or **SHA_CRYPT_MAX_ROUNDS** values is set, then this value will be used.

If **SHA_CRYPT_MIN_ROUNDS** > **SHA_CRYPT_MAX_ROUNDS**, the highest value will be used.

Note: This only affect the generation of group passwords. The generation of user passwords is done by PAM and subject to the PAM configuration. It is recommended to set this variable consistently with the PAM configuration.

SULOG_FILE (string)
If defined, all su activity is logged to this file.

SU_NAME (string)
If defined, the command name to display when running su -. For example, if this is defined as su then a ps will display the command is -su. If not defined, then ps would display the name of the shell actually being run, e.g. something like -sh.

SUB_GID_MIN (number), **SUB_GID_MAX** (number), **SUB_GID_COUNT** (number)
If /etc/subuid exists, the commands **useradd** and **newusers** (unless the user already have subordinate group IDs) allocate **SUB_GID_COUNT** unused group IDs from the range **SUB_GID_MIN** to **SUB_GID_MAX** for each new user.

The default values for **SUB_GID_MIN**, **SUB_GID_MAX**, **SUB_GID_COUNT** are respectively 100000, 600100000 and 10000.

SUB_UID_MIN (number), **SUB_UID_MAX** (number), **SUB_UID_COUNT** (number)
If /etc/subuid exists, the commands **useradd** and **newusers** (unless the user already have subordinate user IDs) allocate **SUB_UID_COUNT** unused user IDs from the range **SUB_UID_MIN** to **SUB_UID_MAX** for each new user.

The default values for **SUB_UID_MIN**, **SUB_UID_MAX**, **SUB_UID_COUNT** are respectively 100000, 600100000 and 10000.

SYS_GID_MAX (number), **SYS_GID_MIN** (number)
Range of group IDs used for the creation of system groups by **useradd**, **groupadd**, or **newusers**.

The default value for **SYS_GID_MIN** (resp. **SYS_GID_MAX**) is 101 (resp. **GID_MIN**-1).

SYS_UID_MAX (number), **SYS_UID_MIN** (number)
Range of user IDs used for the creation of system users by **useradd** or **newusers**.

The default value for **SYS_UID_MIN** (resp. **SYS_UID_MAX**) is 101 (resp. **UID_MIN**-1).

SYSLOG_SG_ENAB (boolean)
Enable syslog logging of **sg** activity.

SYSLOG_SU_ENAB (boolean)
Enable syslog logging of **su** activity - in addition to sulog file logging.

TTYGROUP (string), **TTYPERM** (string)
The terminal permissions: the login tty will be owned by the **TTYGROUP** group, and the permissions will be set to **TTYPERM**.

By default, the ownership of the terminal is set to the users primary group and the permissions are set to *0600*.

TTYGROUP can be either the name of a group or a numeric group identifier.

If you have a **write** program which is setgid to a special group which owns the terminals, define TTYGROUP to the group number and TTYPERM to 0620. Otherwise leave

TTYGROUP commented out and assign TTYPERM to either 622 or 600.

TTYTYPE_FILE (string)
> If defined, file which maps tty line to TERM environment parameter. Each line of the file is in a format something like vt100 tty01.

UID_MAX (number), **UID_MIN** (number)
> Range of user IDs used for the creation of regular users by **useradd** or **newusers**.

> The default value for **UID_MIN** (resp. **UID_MAX**) is 1000 (resp. 60000).

UMASK (number)
> The file mode creation mask is initialized to this value. If not specified, the mask will be initialized to 022.

> **useradd** and **newusers** use this mask to set the mode of the home directory they create

> It is also used by **pam_umask** as the default umask value.

USERDEL_CMD (string)
> If defined, this command is run when removing a user. It should remove any at/cron/print jobs etc. owned by the user to be removed (passed as the first argument).

> The return code of the script is not taken into account.

> Here is an example script, which removes the users cron, at and print jobs:

> #! /bin/sh

> # Check for the required argument.
> if [$# != 1]; then
> echo Usage: $0 username
> exit 1
> fi

> # Remove cron jobs.
> crontab -r -u $1

> # Remove at jobs.
> # Note that it will remove any jobs owned by the same UID,
> # even if it was shared by a different username.
> AT_SPOOL_DIR=/var/spool/cron/atjobs
> find $AT_SPOOL_DIR -name [ˆ.]* -type f -user $1 -delete ;

> # Remove print jobs.
> lprm $1

> # All done.
> exit 0

USERGROUPS_ENAB (boolean)
> If set to *yes*, **userdel** will remove the users group if it contains no more members, and **useradd** will create by default a group with the name of the user.

CROSS REFERENCES

The following cross references show which programs in the shadow password suite use which parameters.

chfn
> CHFN_RESTRICT

chgpasswd
> ENCRYPT_METHOD MAX_MEMBERS_PER_GROUP MD5_CRYPT_ENAB

SHA_CRYPT_MAX_ROUNDS SHA_CRYPT_MIN_ROUNDS

chpasswd
SHA_CRYPT_MAX_ROUNDS SHA_CRYPT_MIN_ROUNDS

gpasswd
ENCRYPT_METHOD MAX_MEMBERS_PER_GROUP MD5_CRYPT_ENAB
SHA_CRYPT_MAX_ROUNDS SHA_CRYPT_MIN_ROUNDS

groupadd
GID_MAX GID_MIN MAX_MEMBERS_PER_GROUP SYS_GID_MAX SYS_GID_MIN

groupdel
MAX_MEMBERS_PER_GROUP

groupmems
MAX_MEMBERS_PER_GROUP

groupmod
MAX_MEMBERS_PER_GROUP

grpck
MAX_MEMBERS_PER_GROUP

grpconv
MAX_MEMBERS_PER_GROUP

grpunconv
MAX_MEMBERS_PER_GROUP

login
CONSOLE_GROUPS DEFAULT_HOME ERASECHAR FAIL_DELAY FAKE_SHELL
HUSHLOGIN_FILE KILLCHAR LOGIN_RETRIES LOGIN_TIMEOUT LOG_OK_LOGINS
LOG_UNKFAIL_ENAB TTYGROUP TTYPERM TTYTYPE_FILE
USERGROUPS_ENAB

newgrp / sg
SYSLOG_SG_ENAB

newusers
ENCRYPT_METHOD GID_MAX GID_MIN MAX_MEMBERS_PER_GROUP
MD5_CRYPT_ENAB PASS_MAX_DAYS PASS_MIN_DAYS PASS_WARN_AGE
SHA_CRYPT_MAX_ROUNDS SHA_CRYPT_MIN_ROUNDS SUB_GID_COUNT
SUB_GID_MAX SUB_GID_MIN SUB_UID_COUNT SUB_UID_MAX SUB_UID_MIN
SYS_GID_MAX SYS_GID_MIN SYS_UID_MAX SYS_UID_MIN UID_MAX UID_MIN
UMASK

pwck
PASS_MAX_DAYS PASS_MIN_DAYS PASS_WARN_AGE

pwconv
PASS_MAX_DAYS PASS_MIN_DAYS PASS_WARN_AGE

su
CONSOLE_GROUPS DEFAULT_HOME ENV_PATH ENV_SUPATH SULOG_FILE
SU_NAME SYSLOG_SU_ENAB

sulogin
ENV_HZ

useradd
CREATE_HOME GID_MAX GID_MIN MAIL_DIR MAX_MEMBERS_PER_GROUP
PASS_MAX_DAYS PASS_MIN_DAYS PASS_WARN_AGE SUB_GID_COUNT
SUB_GID_MAX SUB_GID_MIN SUB_UID_COUNT SUB_UID_MAX SUB_UID_MIN
SYS_GID_MAX SYS_GID_MIN SYS_UID_MAX SYS_UID_MIN UID_MAX UID_MIN

UMASK

userdel
>   MAIL_DIR MAIL_FILE MAX_MEMBERS_PER_GROUP USERDEL_CMD
>   USERGROUPS_ENAB

usermod
>   MAIL_DIR MAIL_FILE MAX_MEMBERS_PER_GROUP

BUGS

Much of the functionality that used to be provided by the shadow password suite is now handled by PAM. Thus, /etc/login.defs is no longer used by **passwd(1)**, or less used by **login(1)**, and **su(1)**. Please refer to the corresponding PAM configuration files instead.

SEE ALSO

login(1), **passwd(1)**, **su(1)**, **passwd(5)**, **shadow(5)**, **pam(8)**.