

Protecting GRUB with a Password

GRUB Project

Installation

```
sudo vi "/usr/local/sbin/grub-set-password"
sudo chmod 0755 "/usr/local/sbin/grub-set-password"
```

```
#!/bin/sh
set -eu

AUTH_CFG="/boot/grub/auth.cfg"

if [ "$(id -u)" -ne 0 ]; then
    echo "This script must be run as root." >&2
    exit 1
fi

printf 'Enter password: '
stty -echo
read -r password1
stty echo
printf '\n'

printf 'Confirm password: '
stty -echo
read -r password2
stty echo
printf '\n'

if [ "$password1" != "$password2" ]; then
    echo "Passwords do not match, aborting." >&2
    exit 1
fi

# Empty password (confirmed twice) disables protection entirely.
if [ -z "$password1" ]; then
```

```

if [ -f "$AUTH_CFG" ]; then
    rm -f "$AUTH_CFG"
    echo "Empty password entered – GRUB password protection disabled."
else
    echo "Empty password entered – nothing to do."
fi
exit 0
fi

hash=$(printf '%s\n%s\n' "$password1" "$password1" \
| grub-mkpasswd-pbkdf2 \
| sed -n 's/^\.*is \ (grub\.pbkdf2\.*\)$/\1/p')

if [ -z "$hash" ]; then
    echo "Failed to generate password hash." >&2
    exit 1
fi

umask 077
cat > "$AUTH_CFG" <<EOF
GRUB_PASSWORD=$hash
EOF
chmod 0600 "$AUTH_CFG"

echo "GRUB password set. Run 'update-grub' to apply it."

```

```

sudo vi "/etc/grub.d/09_password"
sudo chmod 0755 "/etc/grub.d/09_password"

```

```

#!/bin/sh
exec tail -n +3 $0

if [ -f $prefix/auth.cfg ]; then
    source $prefix/auth.cfg
    if [ -n "${GRUB_PASSWORD}" ]; then
        set superusers="admin"
        password_pbkdf2 admin ${GRUB_PASSWORD}
        menuentry_id_option="--unrestricted $menuentry_id_option"
    fi
fi

```

Configuration



Having submenu poses a security risk as it allows bypassing password protection on sub-items.

```
GRUB_DISABLE_SUBMENU=true
```

Usage

```
sudo grub-set-password
```

```
sudo update-grub
```

Revision #26

Created 2026-08-19 19:51:06 UTC by Tomáš Havlas

Updated 2026-09-04 13:17:42 UTC by Tomáš Havlas