

Re: Unix Password Encryption Procedure

Source: <http://unix.derkeiler.com/Newsgroups/comp.unix.programmer/2004-09/1089.html>

From: rc (rc_at_networkz.ch)

Date: 09/28/04

Date: 28 Sep 2004 04:07:56 -0700

kushal.agarwal@gmail.com (Kushal Agarwal) wrote in message
news:<e9d0a198.0409271240.1569a6c9@posting.google.com>...

> Hello,

>

> I know that most Unix machines either use the DES encryption algorithm
> or the MD5 encryption algorithm, I am wondering if there is any
> flavour of unix which uses the kerberos (or anyother) methodology?

On most modern systems, this kind of stuff is controlled by PAM (see
man pam). Using kerberos is only a matter of plugging a suitable
module into the stack, eg on Solaris the stack looks like that:

```
rc@ddp02:~ $ egrep '^login|#login' /etc/pam.conf
login auth requisite pam_authok_get.so.1
login auth required pam_dhkeys.so.1
login auth required pam_unix_auth.so.1
login auth required pam_dial_auth.so.1
# Support for Kerberos V5 authentication (uncomment to use Kerberos)
#login auth optional pam_krb5.so.1 try_first_pass
```

Your questions about MD5 indicates that you might be using Linux, so
maybe here's a good read for you:

<http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/>

>

> Additonally, I know that the function crypt() is able to encrypt using
> either the DES or the MD5 algorithm, depending on the salt supplied
> with the function. I am curious as to given an encrypted string, is
> there any "clean" (via a function(s)) way to determine what method was
> used to encrypt the original string. I need to know how the original
> string was encrypted so that I can use the same procedure to encrypt
> the entered string (so that I may compare the stored and entered
> strings).

>

The MD5 encryption is usually handled by a GNU extension in the crypt
library.

The section GNU EXTENSION in the crypt manpage on linux says:

comp.unix.programmer: Re: Unix Password Encryption Procedure

"If salt is a character string starting with the three characters "\$1\$" followed by at most eight characters, and optionally terminated by "\$", then instead of using the DES machine, the glibc crypt function uses an MD5-based algorithm..."

So if your encrypted string starts wi