EncFS - Transparentní šifrování adresáře s FUSE

V dnešním článku bych vás rád seznámil s několika nejjednodušeji aplikovatelnými (ale samozřejmě účinnými a profesionálními) metodami šifrování, dozvíte se například jak připojit šifrovaný obraz disku jako znakové zařízení či disk, nebo jak šifrovat libovolný adresář, to vše "on-the-fly" (za letu nebo také transparentně). Bezpečnost především…

Úvod

Ačkoli každý z nás občas měl nějaká data, které chtěl skrýt (ať už ve správném a OPODSTATNĚNÉM patologickém schizofrenním paranoidním strachu před black-haty, veřejností, nebo policií), málo kdo se nějakým způsobem zabýval pokusem o jejich šifrování. Musím upozornit, ze těm, kdo potřebují na svém disku ukrývat nějaké věci, před svým nejbližším (převážně Linuxově negramotným) okolím, dostatečně postačí dát na začátek názvu adresáře tečku: <tt>mv pr0n .mami_sem_mi_nelez</tt> ;D, ale protože tady se zabýváme obraně před orgány s mnohem sofistikovanějšími technologiemi a znalostmi než má pravděpodobně někdo, kdo by mohl ve vašem počítači hledat materiály této povahy…

Přehled různých nástrojů

název/popis/algoritmy/deb/další/další zdroje

EncFS

- připojí zašifrovaný adresář do specifikovaného mountpointu, je postavený na FUSE
 - AES, Blowfish, Blowfish-Compat
 - ano (encfs + fuse, libfuse2, rlog, OpenSSL)
 - já po instalaci musel ještě kompilovat FUSE ze staženého tarrballu (naleznete na oficiálním webu), pomocí balíčku libpam-encfs lze připojovat disky automaticky při přihlášení pomocí PAMu
 - Oficialní web

TrueCrypt

- připojí zašifrovaný obraz disku do specifikovaného mountpointu
 - AES, Blowfish, CAST5 (CAST-128), Serpent, Triple DES, Twofish, kombinace předešlých
 - ne
 - Velikost zašifrovaného disku se nedá změnit, umožňuje brute-force útok (alespoň u slabých hesel), existuje verze pro Windows (pouze GUI) a přenosná pro flashdisky…
 - truecrypt.org, root.cz: úvod, více

CryptoLoop (LoSetup)

- Kryptografický wrapper mezi dvě znaková zařízení (nebo znakové zařízení a soubor)
 - AES, XOR, (lze doplnit další, např. DES)
 - ano - já (Etch - 1CD verze), jsem ho měl již předinstalovaný
 - zastaralý, jednoduchý
 - Encryption HOWTO, man losetup
 eCryptFS (SF) [Doplněno 1.11. 2oo7], systém ne nepodobný EncFS. Připojení oddílu ale probíhá takto:

 <tt># mount -t ecryptfs ~/crypt ~/crypt</tt>
 pro používání bez roota potřebujete balíčky ecryptfs-userspace a ecryptfs-utils, pak se pravděpodobně opět mountuje přes dodanou utilitku pomocí FUSE. eCryptFS se sám označuje jako "An Enterprise-class Cryptographic Filesystem for Linux" a patří do rodiny GNUPGFS. S tímto systémem nemám žádné zkušenosti a uvádím ho jen pro úplnost.

 Další informace
 - Také jsou dostupné další kvalitní systémy jako dm-crypt a LUKS, nebo můžete šifrovat jednotlivé soubory nějakou jednoduchou utilitkou... Tyto možnosti jsem ale již tolik nerozebíral (vzhledem ke své spokojenosti s EncFS).
 - Wikipedia/List_of_cryptographic_file_systems
 - Wikipedia/Full_disk_encryption

 <h3>Vše závisí jen na vašich konkrétních požadavcích a stupni důvěrnosti dat.</h3>

EncFS

Tento program jsem si vybral já, protože je nejpohodlnější a třeba narozdíl od také velmi kvalitního TrueCryptu má dynamickou velikost šifrovaného disku a tak nemusím řešit problémy s příliš malým, nebo příliš velkým diskem... Samozřejmě, že není pozadu ani co se týče bezpečnosti.

 K připojení slouží program encfs: <tt>encfs ~/.crypt ~/crypt</tt>
 První argument je adresář se zašifrovanými daty, druhý je ten, do kterého se připojí data dešifrovaná - pro čtení a zápis. Dešifrovaná data tedy nejsou fyzicky na disku a tedy se k nim nikdo jentak nedostane...

 Při prvním spuštění budou zadané adresáře vytvořeny (pokud již neexistují), potom budete mít na výběr (nejlepším výběrem pro nás bude zadání znaku "p" - paranoidní nastavení (AES-256), zkušenější mohou dát "x" - vlastní nastavení, ovšem pokud nevíte jaký je rozdíl mezi různými algoritmy a jejimi obdobami, verzemi a jak má vypadat bezpečný klíč pro daný algoritmus, nebo prostě nemáte náladu nic řešit, použijte raději "p"). Na druhou stranu v "x" režimu lze například vypnout šifrování názvů souborů (nebezpečné!!! - útočník může zaměnit soubor, ke kterému nemá přístup s tím, který mu poskytujete dobrovlně) a spousta dalších nastavení.

 Pak budete dotázáni na heslo (poprvé ho zadejte ještě jednou pro ověření)...

 Dešifrovaný adresář odpojíte příkazem <tt>fusermount -u ~/crypt</tt> (kde adresář je mountpoint - tedy rozšifrovaná data)

 Šťouralové si jistě všimnou, že v adresáři se zašifrovanými daty je soubor .encfs? (? = zřejmě číslo podverze - v mém případě EncFS 1.2.5 má soubor .encfs5), který obsahuje všecna nastavení šifrování našeho adresáře. Já dopručuji tento soubor pečlivě zazálohovat stejně dobře, jako si pamatujete heslo, je totiž pravděpodobné, že v případě jeho poškození přijdete o všechna zašifrovaná

data!!!

 [Doplněno 1.11. 2oo7]: EncFS obsahuje ještě dvě další utilitky, tou první je <tt>encfssh</tt>, která udělá pouze to, že vytvoří dočasný zašifrovaný adresář a otevře nový shell, jakmile ukončíte svojí práci v tomto shellu (např. Ctrl+D), tento adresář se smaže. Ve skutečnosti jde pouze o jednoduchý skript.

 Další utilitkou je <tt>encfsctl</tt>, ta vám umožní provádět různé změny v nastavení a jiné pomůcky (změna hesla, jednorázový přístup k zašifrovaným souborům, atd..). Pro více informací viz. screenshot.

 <h2>Screenshot z encfsctl:</h2>

```
harvie@harvie-srv:~$ encfsctl<br />
encfsctl version 1.2.5<br />
Usage:<br />
encfsctl (root dir)<br />
  -- displays information about the filesystem, or<br />
encfsctl info (root dir)<br />
  -- zobraz informace (Implicitní příkaz)<br />
encfsctl passwd (root dir)<br />
  -- změnit heslo pro svazek<br />
encfsctl autopasswd (root dir)<br />
  -- change password for volume, taking password from standard input.<br />
        No prompts are issued.<br />
encfsctl showcruft (root dir)<br />
  -- ukázat nerozkódovatelná jména souborů ve svazku<br />
encfsctl cat (root dir) path<br />
  -- decodes the file and cats it to standard out<br />
encfsctl decode (root dir) encoded-name<br />
  -- rozkódovat jméno a ukázat jeho čitelnou verzi<br />
encfsctl encode (root dir) [plaintext-name]<br />
  -- encodes a filename and print result<br />
encfsctl export (root dir) path<br />
  -- decrypts a volume and writes results to path<br />
encfsctl --version<br />
  -- vypiš číslo verze a ukonči běh<br />
<br />
Example:<br />
encfsctl info ~/.crypt<br />
<br />
harvie@harvie-srv:~$ encfsctl info ~/.crypt<br />
<br />
Version 5 configuration; created by EncFS 1.2.5 (revision 20040813)<br />
Filesystem cipher: "ssl/aes", version 2:1:1<br />
Filename encoding: "nameio/block", version 3:0:1<br />
Key Size: 256 bits<br />
Block Size: 512 bytes, including 8 byte MAC header<br />
Each file contains 8 byte header with unique IV data.<br />
Filenames encoded using IV chaining mode.<br />
File data IV is chained to filename IV.<br />
<br />
harvie@harvie-srv:~$
```


 <h2>A na závěr screenshot z expertního režimu encfs:</h2>

```
harvie@harvie-ntb:~$ encfs ~/.crypt ~/crypt<br />
The directory "/home/harvie/.crypt/" does not exist. Should it be created?
(y,n) y<br />
The directory "/home/harvie/crypt/" does not exist. Should it be created?
(y,n) y<br />
Creating new encrypted volume.<br />
Please choose from one of the following options:<br />
 enter "x" for expert configuration mode,<br />
 enter "p" for pre-configured paranoia mode,<br />
 anything else, or an empty line will select standard mode.<br />
?> x<br />
<br />
Manual configuration mode selected.<br />
The following cipher algorithms are available:<br />
1. AES : 16 byte block cipher<br />
 -- Supports key lengths of 128 to 256 bits<br />
 -- Supports block sizes of 64 to 4096 bytes<br />
2. Blowfish : 8 byte block cipher<br />
 -- Supports key lengths of 128 to 256 bits<br />
 -- Supports block sizes of 64 to 4096 bytes<br />
3. blowfish-compat : algorithm compatible with EncFS 0.2-0.6<br />
 -- key length 160 bits<br />
 -- block size 64 bytes<br />
<br />
Enter the number corresponding to your choice: 1<br />
<br />
Selected algorithm "AES"<br />
<br />
Please select a key size in bits.  The cipher you have chosen<br />
supports sizes from 128 to 256 bits in increments of 64 bits.<br />
For example:<br />
128, 192, 256<br />
Selected key size: 256<br />
<br />
Using key size of 256 bits<br />
           ---------------- Zkráceno ----------------<br />
```


 THX2Ihrisko.org (EncFS quickstart by Wiro Wire)