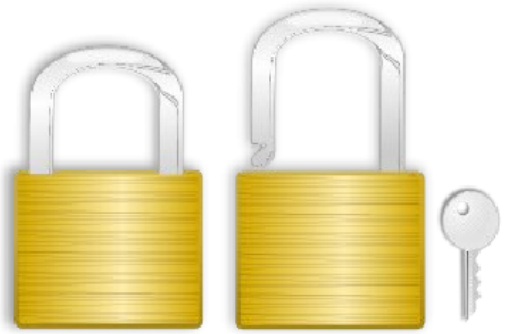
Titkosítsd fájljaidat zuluCrypt-tel

Írta: muungwana

A ZuluCrypt [1] olyan projekt, ami megpróbál egyszerűbb kezelő felületet adni a cryptsetup-nak [2] és a tcplay-nek [3]. Tcplay egy parancssori eszköz, ami képes TrueCrypt (szabad) formátumú titkosított köteteket készíteni. A Cryptsetup parancssori eszköz, ami LUKS-ra formázott titkosított kötetek készítésére és megnyitására használható. A Cryptsetup képes TrueCrypt kötetek megnyitására.

Kétfajta titkosított kötet létezik. Vannak olyanok, amik használnak valamit, amit „fejlécként” ismerünk és vannak, amik nem. A TrueCrypt és a LUKS kötetek alkalmaznak fejlécet. A Cryptsetup-nak „sima dmcrypt”-nek ismert titkosítási formátuma van, ami nem használ fejlécet.

A titkosított kötetek kétfajta fejlécet használnak. Vannak titkosított fejlécet alkalmazók és vannak amik nem alkalmaznak. A TrueCrypt fejléce titkosított, míg a LUKS-é nem. A titkosítatlan fejlécet használó LUKS nyilvánvalóvá teszi mindenki számára, hogy a kötet egy LUKS titkosított kötet és ez egyesek számára problémás lehet.



Ahogy a TrueCrypt-ben is használják, a titkosított fejléc, vagy a fejléc hiánya, mint a PLAIN-ben (sima dmcrypt), megkülönböztethetetlenné teszik a kötetet a véletlenszerű zajtól és ez első pillantásra hasznos lehet. Ugyanakkor ez hasznosság gyanú esetén nem segít, mivel nem nagy az esélye annak, hogy bárki elhiszi egy 100 GB-os, véletlen zajnak tűnő titkosított fájlról, hogy az csak egy véletlenszerű adatfájl és nem egy titkosított kötetet tartalmazó konténer fájl. Ezek a „kézenfekvő tagadás”-ként ismert fejléc nélküli és rejtett fejléces kötetek, jelenleg is vita tárgyát képezik a titkosítók körében.

Többek között a TrueCrypt, a LUKS, az Apple's FileVault, és a Microsoft's BitLocker is magában a fejlécben tárolja a kötet megfejtéshez szükséges információkat. Ezeknél a köteteknél szükség van a fejlécre ahhoz, hogy a kötet megnyitható legyen és egy hiányzó, vagy sérült fejléc elérhetetlenné teszi a kötet megnyitását. Nagyon fontos legalább egy fejlécmásolat biztonságos helyen való tárolása arra az esetre, ha a kötetek valamelyike megsérül valahogy.

LUKS a „Linux Unified Key Setup” rövidítése. Ez egy, a LUKS formátumú titkosított kötetek megnyitását lehetővé tévő, az információk tárolásának módját leíró specifikáció. A LUKS titkosítási formátum a Linux-ban szabvány és használata Linux rendszerekben ajánlott, titkosított kötet alkalmazása esetén. A TrueCrypt jobb választás, ha a titkosított kötetet Linux, Windows és OSX számítógépek között kell megosztani.

A ZuluCrypt háromféle titkosított kötetet képes készíteni és megnyitni, LUKS-ot, TrueCrypt-et és PLAIN-t. A PLAIN-kötet fejléc nélküli, kevéssé titkosított, mivel az összes titkosítási információt a zuluCrypt állítja elő amikor készíti, vagy megnyitja ezeket a köteteket.

A háromféle kötet előnyei és hátrányai

PLAIN:

Előnyök:

Első, nem használ kötetfejlécet, ezért nem lehet „lefagyasztani” az egész kötetet egy kis részének egyszerű felülírásával. Második, nem használ fejlécet, ezért nem lehet megtudni, hogy a kötet valójában titkosítottnak tűnő véletlen adatot, vagy titkosított kötetet tartalmaz.

Hátrányok:

Nem használ fejlécet, ezért minden eszköznek, ami megnyithatja ezeket a köteteket, rendelkeznie kell azzal a titkosítási opcióval, amivel készítették. Eltérő eszközök, eltérő titkosítási opciókat tartalmaznak, ezáltal az ilyen kötetek nem nagyon mozgathatók alkalmazások között, vagy éppen az adott alkalmazás más verziói között.

TrueCrypt

Előnyök:

Első, titkosított fejlécet használ, így nem tudható, hogy a kötet TrueCrypt formátumú titkosított kötet, vagy a kötet titkosítottnak tűnő, véletlenszerű adatokat tartalmaz. Második, hogy ez rejtett kötet. A TrueCrypt kötet akár két különböző titkosított kötetet is tartalmazhat. Az első kötet népszerű nevén „a másik kötet" és a második opcionális pedig „rejtett kötet"-ként ismert. Amikor egy TrueCrypt kötetet megnyitunk, a felhasználó – megfelelő kulcs megadásával – kiválaszthatja, hogy a kettő közül melyiket nyissa ki.

Hátrányok:

Fejlécet használ. Lehetetlen megnyitni titkosított fejlécet alkalmazó kötetet a fejléce nélkül és ezáltal egy sérült TrueCrypt fejléc elérhetetlenné teszi a kötet megnyitását. Ha TrueCrypt kötetet használsz, gondoskodj arról, hogy legalább egy másolatod legyen a kötet fejlécéről.

LUKS

Előnyök:

A LUKS-kötet összesen 8 különböző kulccsal nyitható meg.

Hátrányok:

Első, a LUKS fejléc nyíltan tárolt, egyértelművé téve, hogy ez egy LUKS formátumú titkosított kötet és így bizonyos körülmények között nem kívánatos. Lehet készíteni LUKS kötetet csatolt fejléccel és a zuluCrypt képes megnyitni ezeket a köteteket LUKS kiterjesztés használatával. Második, fejlécet használ. Nem lehet megnyitni fejlécet használó titkosított kötetet a fejléc nélkül, ezért egy hibás LUKS fejléc lehetetlenné teheti a kötet megnyitását. Ha LUKS kötetet használsz, akkor gondoskodj arról, hogy legalább egy másolatod legyen a kötet fejlécéről.

A zuluCrypt kétfajta titkosításra képes. Képes egyetlen fájl titkosítására, megfejtésére, vagy blokk-eszköz titkosításra.

Fájltitkosítás.

A fájltitkosítás a libgcrypt titkosítót használja végső eszközként. A fájlokat 256 bites CBC módú AES-sel titkosítja. A titkosító kulcs a felhasználó jelszavából kivonatolással készül pdkdf2-vel, 10 000 fordulós iterációval és sha2 szolgáltatja a titkosítási töredék funkciót. Az eredményként kapott, titkosított fájl mérete 64+1024*n byte méretű, ahol az n egy 0-tól kezdődő szám.

A fájltitkosítási funkció azoknak való, akik egy, vagy két fájlt akarnak titkosítva tartani, de nem szeretnek bajlódni titkosított tárolók kezelésével képfájlokban. Ez a funkció kicsit olyan, mint a gpg fájltitkosítás használata szimmetrikus kulccsal.

Hogyan készítsünk titkosított fájlt:

1. Indítsd el a zuluCrypt-et.

2. Lépj be a menübe és kattints a „zC → Encrypt a file"-ra, hogy nyiss egy titkosítási párbeszédablakot.

3. A megjelenő párbeszédben kattints a „source path" szöveggel azonos sorban lévő gombra. A fájlpárbeszéd jelenik meg. Válaszd ki a titkosítani szándékozott fájlt és írd be a fájltitkosításra használt jelszót, majd kattints a „Create" (elkészít) gombra. A fájl titkosított változatát a „destination path"-ban meghatározott helyen hozza létre.

Az előbbi módon készült fájl kibontása:

1. Indítsd el a zuluCrypt-et.

2. Menj a menübe és kattints a „zC → decrypt a file"-ra, hogy megnyíljon a megfejtés ablak.

3. A párbeszédben kattints arra a gombra, ami a „source path" szöveggel egy sorban van. Egy fájlpárbeszéd jelenik meg, és válaszd ki a kibontani szánt fájlt. Írd be a titkosításra használt jelszót, majd kattints a „Create"-re. A fájl visszafejtett változatát a „destination path"-ban meghatározott helyen hozza létre.

Blokkeszköz titkosítása

Egy merevlemez, vagy egy USB-kulcs két példa a blokkeszközökre. Egy szokásos fájl szimulálhat blokkeszközt egy „loop devices"-nak hívott eszközzel. Ezek az eszközök „/dev/loop"-pal kezdődő útvonalat kapnak.

A Linux kernelben a blokkeszközökkel foglalkozó infrastruktúrát „dmcrypt"-nek hívják és a feladatait OTF-ként (on the fly – röptében) ismert folyamat

formájában végzi. Dmcrypt eszközöket a „/dev/dm-”-mel kezdődő eszközcímek jelzik és ezek az útvonalak általában a „/dev/mapper”-ben található szimbolikus hivatkozáson keresztül érhetők el.

A következő egy példa egy 100 MB-s titkosított tároló létrehozására egy fájlban és egy fájl hozzáadására a biztonságos tárolás érdekében.

1. Hozz létre egy 100 MB-s fájlt.

2. Csatolj egy loop eszközt a fájlhoz.

3. Készíts egy OTF titkosítási „mapper”-t a loop eszközre.

4. Készíts egy fájlrendszert a titkosítási mapper-re

5. Csatold a fájlrendszert a mapper-re.

6. Másold a biztonságosan tárolni szándékozott fájlt a fájlrendszerre a csatolási ponton keresztül.

7. Válaszd le a fájlrendszert.

8. Semmisítsd meg az OTF titkosítási mapper-t.

9. Válaszd le a loop eszközt a fájlról.

10. Kezeld a titkosított kötetet úgy, mint egy biztonságos fájltárolót.

A zuluCrypt grafikus felületet biztosít a fentiekben meghatározott feladatok könnyű végrehajtásához.

Az előbbi lépésekkel:

Az első lépés egy „/home/ink/secret.img” szerű útvonallal dolgozik. Ez egy útvonal egy szokásos fájlhoz.

A második konvertálja a „/home/ink/secret.img” fájlt valami a „/dev/loop0” loop eszközhöz hasonlóra.

A harmadik lépés konvertálja a „/dev/loop0”-t valami „/dev/mapper/secrets.img”-re. A „/dev/mapper/secrets.img”-be írt adat titkosítva lesz és úton a „/home/ink/secret.img” felé átkerül a „/dev/loop0”-ra. A „/dev/mapper/secrets.img”-ből vett adatokat a „/dev/loop0” veszi át, onnan pedig a „/home/ink/secret.img” olvas, amit a dmcrypt fejti vissza és ad ki a felhasználónak. Ezt a folyamatot nevezik „on the fly” (röptében) titkosításnak, mivel a titkosítási mapper nem tárol, vagy tart adatot. Ehelyett inkább kapja az adatot, titkosítja, vagy megfejti az adatáramlás irányától függően, majd továbbadja.

Hogyan készítsünk titkosított tárolót egy képfájlban.

1. Indítsd el a zuluCrypt-et.

2. Menj a menüben a „Create → Encrypted container in a file”, hogy egy párbeszédablakot nyiss.

3. Írd be annak a fájl nevét a „file name” mezőbe, ami majd tartalmazza a tárolót.

4. Írd be a konténer méretét a „file size” mezőbe.

5. Kattints a „Create”-re.

6. Várd meg amíg a konténerfájl létrejön és a megjelenik a kötetkészítési párbeszédablak.

7. Írd be a kötet létrehozásához használt jelszót.

8. Válaszd ki a létrehozni tervezett kötet típusát a „volume type” listából.

9. Kattints a „Create”-re a kötet létrehozásához.

Hogyan készíts titkosított tárolót egy partíción.

1. Indítsd el a zuluCrypt-et.

2. Menj a Create → Encrypted container in a partition-ra egy párbeszédablak megnyitásához.

3. Kattints duplán a kötetnek szánt partícióra, majd menj a 7. pontra az előző listában. Ha a partíció, amiben titkosított konténert akarsz elhelyezni nem jelenik meg a listában, akkor indítsd újra a zuluCrypt-et rendszergazda felhasználóként és próbáld újra.

Hogyan nyiss meg egy fájlban elhelyezett titkosított konténert zuluCrypt-tel.

1. Indítsd el a zuluCrypt-et.

2. A menüben menj az „Open → Encrypted container in a file”-ra egy párbeszéd megnyitásához.

3. A párbeszédben kattints a „volume path” mezőtől jobbra található gombra és ezután menj oda, ahol a kötet található, és kattints rá a megnyitáshoz. Alternatívaként egyszerűen ragadd meg a kötetfájlt a zuluCrypt-ben, és jelszót létrehozó párbeszéd jelenik meg, amibe az útvonal leírása már bekerült.

4. Írd be a kötet kulcsát a „volume key” mezőbe és kattints az „Open”-re a kötet megnyitásához.

Hogyan nyissunk meg egy partíción található titkosított konténert zuluCrypt-tel.

1. Indítsd a zuluCrypt-et.

2. Menj a menüben az „Open → Encrypted container in a partition”-ra, hogy megnyiss egy párbeszédablakot.

3. A párbeszédben kattints, illetve duplán kattints a megnyitandó, titkosított kötetet tartalmazó partíción.

4. Írd be a kötet kulcsát a „volume key” mezőbe és kattints az „Open”-re a kötet megnyitásához.

Mindkét előbb említett módszernél a kötet megnyílik és csatolódik az útvonalhoz, aminek az utolsó elemét a „mount name” mező határozza meg. Amikor a kötet sikerrel csatolódik, a zuluCrypt automatikusanmegnyitja a csatolási pont útvonalát. A kötet bezárásához kattints a zuluCrypt ablakban szereplő bejegyzésére és kattints a „Close”-ra a megjelenő ablakban.

A zuluCrypt képes megnyitni titkosított kötetet más forrásból származó kulccsal. Ez a forrás többek között lehet jelszó; kulcsfájl; a kwallet-ből kinyert kulcs; a Gnome libsecret-ből származó kulcs; belső, biztonságos tárolórendszerből származó kulcs és GPG-val titkosított kulcsfájl.

Jelszó kötetkulcsként történő használathoz gondoskodj, hogy a kulcs forrás opció olvassa a „kulcsot” és ezután írd be a jelszót a beviteli mező alján.

Kulcsfájl használatához kötetkulcs forrásaként, kattints az opciós sávon, válaszd a „keyfile”-t és nyomd meg a jobbra lent található gombot, ami lehetővé teszi a kulcsfájl helyének megkeresését.

Kiterjesztés kötetkulcskénti használatához az opciós sávon válaszd a „plugin”-t, majd nyomd le a jobbra lent látható gombot. Az elérhető kiterjeszések listája megjelenik. Válaszd ki a megfelelőt a listból.

A Kwallet-ben, a Gnome kulcskarikán, vagy biztonságos belső tárolórendszer kiterjesztésében tárolt kötetkulcsot a menü „Options → Manage volumes in internal/kde/gnome wallet”-re kattintva lehet kezelni.

A kulcs tárolására Gnome-nál legjobb a Gnome tárca (wallet), kulcskarika (keyring), de evvel van némi gond. A kulcsokat a felhasználó kulcskarikáján tárolják és ez a kulcskarika kinyitásra kerül, amikor a felhasználó bejelentkezik. Innentől a kulcskarika nyitva, így bármely, a munkamenet során futó alkalmazás olvasni tudja a tárolórendszer által alkalmazott nyilvános API-k alkalmazásával azokat a kulcsokat.

KDE rendszerben a Kwallet biztonsági tárolórendszer tűnik a legmegfelelőbbnek, de ugyanavval a biztonsági problémával küzd, mint a Gnome biztonsági tárolórendszere. Mihelyst a tárcát megnyitották, akkor bármely, a felhasználói munkamenet alatt futó alkalmazás elérheti azokat, a tárolórendszer által alkalmazott nyilvános API-kon keresztül.

A fenti biztonsági tárolórendszerek viselkedése a kialakítás miatt nem lehet ideális egyes felhasználók számára bizonyos esetekben. A belső biztonsági tárolórendszert a libcrypt szolgálja ki és nem hasonlít a viselkedése az előbbi két rendszeréhez. Egy nyitott belső biztonsági tárolórendszer csak az azt megnyitó zuluCrypt példány által érhető el.

Kedvencek

Kényelmi okokból a legtöbb kötet megnyitható úgy is, hogy hozzáadjuk a kedvencek listájához.

Elemek a kedvencek listájához a menüben az „Options → Manage favorites”-re kattintva megnyíló ablakban adhatók hozzá. A Kedvencek elemei a „Favorite”-ra kattintva a menüben adhatók hozzá.

Adat törlése egy meghajtón

Nagyon fontos, hogy titkosítási szempontból erősnek tekinthető adatokra írjuk, lehetetlenné téve annak meghatározását, hogy a kötet mely része van használatban és melyik nincs. Ha a titkosított kötet kitalálható adatmintákat tartalmaz, mint például ha csak 0-kat teleírt eszközre készül, akkor mélyelemzés megállapíthatja, hogy a titkosított kötet mennyire és mely része van használatban.

Amikor eszközön titkosított tárolót készítünk, a zuluCrypt felajánlja, hogy először véletlenszerű adatokkal megtölti az eszközt. Ez a lehetőség más eszközökön a menü „Erase data in a device” aktiválásával hajtható végre. A véletlenszerű adatok a lemezre egy sima dmcrypt titkosító mapper 64 bites véletlenszerű kulcsának az adott eszközön történő megnyitásával kerülnek kiírásra, amit azután 0-kal bombázz a mapper-en keresztül. Ez a technika, más alternatívákhoz képest, mint pl. a /dev/urandom-ból véletlenszerű adatok kiírása, a leggyorsabbnak bizonyult.

Rendszer és nem-rendszer kötetek

Az elérés felügyeletére, azaz a felhasználók és a blokkeszközök mihez férnek hozzá és mit csinálhatnak azzal, amit elérnek, a zuluCrypt egy „rendszerkötet”, „nem-rendszerkötet” koncepciót alkalmaz.

Rendszerkötetnek minősül az, aminek aktív bejegyzése van az „/etc/fstab”-ban, az „/etc/crypttab”-ban, az „/etc/zuluCrypt/system_volume.list”-ben, vagy engedélyezett udev esetén az udev annak azonosítja. Ideális esetben minden számítógépen belüli kötet rendszerkötetnek minősül.

Nem-rendszer kötet az, ami nem felel meg a fenti megkötéseknek, vagy nincs bejegyzése az „/etc/zuluCrypt/non_system_volumes.list”-ben.
Ideális esetben ezek a kötetek bedugható USB alapú merevlemezek, vagy USB-kulcsok.

Partíciók hozzáadhatók, vagy eltávolíthatók a rendszer, vagy nem-rendszer kötetek listájából egyszerűen a zuluCrypt rendszergazdakénti indításával, majd belépve a menü „Options → Manage system volumes/manage non system volumes” pontjába és ezután a kötet hozzáadásával a megfelelő listához.

Engedélyek.

A zuluCrypt a Unix engedélyrendszerével, két csoportot - zulucrypt és zulumount - létrehozva szabályozza a felhasználó jogait a blokkeszközre.

Ha egy eszköz rendszereszköznek minősül, akkor azon csak a rendszergazda, vagy a „zulucrypt”-ben szereplő felhasználó készíthet titkosított kötetet, vagy vehet el, állíthat helyre kötet fejlécet. Ha egy eszközön kötetet akarsz létrehozni és az eszköz nem jelenik meg a listában, indítsd újra a zuluCrypt-et root-ként és próbáld újra.

Ha egy eszköz rendszereszközként lett azonosítva, a zuluMount csak akkor csatolja, ha a felhasználó root és a zulumount csoport tagja, vagy az eszköz rendelkezik bejegyzéssel az „/etc/fstab”-ban „user”, vagy „users” csatolási opcióval beállítva.

ZuluMount.

A zuluMount egy általános célú csatoló eszköz, ami képes megnyitni zuluCrypt által támogatott titkosított köteteket csakúgy, mint a nem titkosítottakat.

A zuluMount képes automatikusan érzékelni a bedugott eszközöket és automatikusan csatolni.

Lábjegyzeti referenciák:

[1] http://code.google.com/p/zulucrypt/

[2] http://code.google.com/p/cryptsetup/

[3] https://github.com/bwalex/tcplay

[4] http://www.truecrypt.org/