**CMPE 455 Güz 2019**

Alexander Chefranov ve Tansel Sarıhan tarafından hazırlanmıştır.

**Linux'ta Erişim Kontrolü**

**içindekiler**

CMPE455 Bahar 2019

Linux'ta Erişim Kontrolü

1. Görev
2. Fedora İşle m Sistemi
	1. Bash Terminali
	2. Bazı Önemli ve Faydalı Komutların Listesi
3. Giriş kontrolüne giriş
4. Unix dosya izinleri ve düğümleri örnek 1 Örnek 2

Örnek 3

Örnek 4

Örnek 5

1. İzinlerin sekizli gösterimi

Örnek 6

1. Linux sistemindeki dosya izinlerini değiş rmek, kullanıcı, grup eklemek, dosya sahibini ve grubunu değiş rmek Örnek 7

Örnek 8

Örnek 9

Örnek 10

Örnek 11

Örnek 12

Örnek 14

Örnek 15

Örnek 16

Örnek 17

Örnek 18

Örnek 19

Örnek 20

Örnek 21

Örnek 22

Örnek 23

Örnek 24

Örnek 25

Mücadelesi 1.

1. Komut umask

Örnek 26

2. meydan okuma.

1. UID'yi (SUID) ayarlayın

Örnek 27

Örnek 28

Örnek 29

Örnek 30

1. C de bir SUID programı yazmak

Örnek 31

Örnek 32

Örnek 33

Örnek 34

Örnek 35

Meydan 3

4. zorluk

Meydan 5

Meydan 6

1. Linux Genişle lmiş ACL'ler Örnek 36
2. Sonuç

Referanslar

O ur laboratuar malzemes göre [1] . Bölüm 1 görev açıklar. Bölüm 2, 3 en öneml Lnux komutlarını ve erşm kontrolü kavramlarını kısaca açıklamaktadır. Fedora Lnux tabanlı şletm sstemyle laboratuarlarda çalışacaksınız. 4-10. Bölümler örnekler (Kal Lnux ( ekranlarda alex @ lenovo le)) ve Fedora ( ekranlarda lnuxlab @ asus le brlkte) şletm sstemlernde hazırlanmış) ve zorluklar çerr. Bölüm 11 malzemey tamamlıyor.

1. **Görev**

Kendi örneklerini çalış ran doğrultusunda aşağıda malzemeyi Oku Örnekler tekrar laboratuar malzemesi gösterilen 1-36 (ekran) ( szn kullanmak kend yerne kullanıcı adını alex veya lnuxlab örn “ chmod 701 / home / kullanıcı\_adınız /”) . Szn tarafınızdan çözülmes beklenen 1-6. Zorluklar var. Laboratuar gruplar halnde yapılacaktır. Her grup, Örnek 1-36, 1-6 Mücadeleler ve açıklamaların varyantlarının ekran görüntülern çeren çalışmalar hakkında br rapor hazırlar.

**Raporlama için vade tarihi 30 Ekim 2019, 16.30'dur. Örneklerinizi ve çözümlerinizi çalıştıracak ve bunları Laboratuar Asistanı'nın sorularını cevaplayarak açıklayacaksınız.**

1. **Fedora İşletim Sistemi**

Fedora, topluluk destekli Fedora Projesi tara ndan geliş rilen bir Linux dağı mıdır. Açık kaynak kodlu ve ücretsiz. F kullanıcıya işbirliğiyle geliş rilen çünkü Edora işle m sistemi kullanıcı dostudur.

## 2.1 Bash Terminali

W e bizim deneyler için sık bash terminalini kullanacak. Bu nedenle, bash terminalini aç ğımızda ne

göreceğimizi öğrenelim.

 

Terminali aç ğımızda yukarıdaki gibi bir görüntü görüyoruz. “ Linuxlab ” "bizim kullanıcı adıdır olarak bize " sonra "@" işare kullandığımız makinenin adıdır ve "~" kullanıcı giriş dizinini anlamına gelir.



Aynı şekilde, “root” bir kullanıcı adıdır ( sistemde tüm yetkilere sahip r), “ asus ” makine adıdır ve “#” yöne ci olarak giriş yap ğımız anlamına gelir (root).

## 2.2 Bazı Önemli ve Faydalı Komutların Listesi

Fedora, dnf paket yöne cisini kullanır . Herhangi bir açık kaynaklı program veya araç yüklemek is yorsanız , dnf kullanmanız gerekir . Örneğin, me nlerinizi veya program kodlarınızı düzenlemek için bir me n düzenleyici kurmanız gerekir. Terminalde “ dnf install kwrite ” komutu kullanılarak “ KWrite ” me n editörü kurulabilir .



Sudo komutu - sudo komutu daha kolay Fedora sistemini yönetmek kolaylaş rır. Fedora'daki bazı komutların yalnızca ayrıcalıklı bir kullanıcı veya yöne ci tara ndan çalış rılmasını bekler . Sudo komutu sağlar çalış rmak sen sanki bir komut kökü olarak bilinen yöne ci. Örneğin, “ sudo chmod 666 somefile ”.



Su komutu - kullanıcıyı değiş r. Örneğin, mevcut kullanıcı ile öğrenci kullanıcı arasında geçiş yapmak için “ su student” komutu kullanılabilir.



Ls komutu - dosyaların ve dizinlerin dizin içeriğini listeler. Örneğin, “ ls / home / linuxlab ” komutu çalış rıldığında, / home / linuxlab dizininin içeriği terminal penceresinde gösterilecek r.

 

C

d

k

o

m

u

t

u

-

g

e

ç

e

r

l

i

d

i

z

i

n

i

d

e

ğ

i

ş

r

i

r

.

Ö

r

n

e

ğ

i

n

,

“

c

d

D

e

s

k

t

o

p

”

k

o

m

u

t

u

g

e

ç

e

r

l

i

d

i

z

i

n

i

/

D

e

s

k

t

o

p

d

i

z

i

n

i

n

e

d

e

ğ

i

ş

r

i

r

.

(

İ

ç

i

n

d

e

a

y

n

ı

ş

e

k

i

l

d

e

,

“

c

d

.

.

”

k

o

m

u

t

u

d

a

b

i

r

ö

n

c

e

k

i

d

i

z

i

n

e

d

ö

n

m

e

k

i

ç

i

n

k

u

l

l

a

n

ı

l

a

b

i

l

i

r

.

)



Mkdir - Bu komut yeni bir dizin oluşturur. “ MyLabWorks ” adında yeni bir dizin oluşturmak için örnek kullanım : “

 

Rm komutu – bu komut bazı dosya veyadizileri silmek için kullanılabilir.F veya örnek, biz silebilir “test.txt” “dosya rm.txt”

 

1. **Giriş kontrolüne giriş**

Erişim kontrolü, hangi işlemlere izin verildiğini belirleyerek ve uygulayarak yetkilendirmeyi zorlar. Bazı terminoloji: özne, kullanıcı veya program gibi eylemleri gerçekleş ren ak f bir varlık r ve nesne, dosya veya ağ kaynağı gibi erişilebilen (genellikle pasif) bir kaynak r. Erişim kontrolü, hangi eylemlere izin verildiğini ve hangilerine izin verilmediğini sınırlayan bir güvenlik poli kası uygulayarak deneklerin nesnelere erişmesine aracılık eder. Poli ka, bir kurallar dizisi olarak resmi veya gayrı resmi olarak izin verilenleri ifade eder. Erişim kontrol m echanism bir poli kası uygular kodu veya şeydir. Erişim kontrolü modeli, bir poli kayı veya poli ka türlerini temsil etme ve akıl yürütme yöntemidir.

1. **Unix dosya izinleri ve düğümleri**

Geleneksel Unx güvenlk model, kullanıcıların “sahp oldukları” kaynaklara kmn erşebleceğn yapılandırmalarını sağlayan steğe bağlı erşm kontrolü (DAC) modeln temel alır. Her kullanıcı, hang kullanıcıların oluşturdukları dosyalara erşebleceğn kontrol edeblr. Bu, kullanıcıların br sstem yönetcsne dahl olmadan zn vermelern sağlar. Bu, geleneksel olarak Wndows ve Unx gb çoğu tüketc şletm sstemne yerleşk br güvenlk türüdür . Unx dosya znler kısaltılmış (bastleştrlmş) br erşm kontrol lstes (ACL) kullanır. Br (tam) ACL, her konunun br lstesn ve her dosyaya ne yapableceklern eklemey çerr (Wndows dosya erşmn bu şeklde yönetr). Örneğn, br dosya bu ACL'ye sahp olablr: “Joe okuyablr, Frank yazablr, Alce okuyablr ve Eve okuyablr”. Unx , bu üç tür konu çn yalnızca kuralları tanımlayarak znler bastleştrr:

* Dosyanın sahb olan kullanıcı (u)
* dosya grubu (g) ● Dğer kullanıcılar (o) ls komutu dosyaların znlern görüntülemek çn kullanılır, -l bayrağı ayrıntılı çıktı sağlar; örneğn, ls –l / bn / ls , ls komutunun znlern görüntüler.
* 

File path

Last modified

Files size (bytes)

File group is root

File owner is root

The file has this many names (hard links)

Others can read and execute

Group can read and execute

Owner can read, write, execute

* **w** : Dosyanın çerğn değştrn
* **x** : Dosyayı br şlem olarak yürütün (İlk brkaç bayt ne tür br çalıştırılablr olduğunu, br program veya komut dosyasını tanımlar )

Br dzn çn (lk sembol d'dr):

* **r** : Dznde hang dosyaların olduğunu görün
* **w** : Rehberden sm ekleyn, yenden adlandırın veya sln
* **x** : 'stat' (dosya sahplern ve boyutlarını görüntüleyn), cd'y dzne yerleştrn ve

çndek dosyalar

* **t** (x yerne), AKA “yapışkan bt”: yazma, dznden br dosyayı slmek çn yeterl değl , bu durumda dosyanın da kendsne at olması gerekr.

Her dosya çn znler dosyanın node'unda saklanır . Br düğüm , br ver yapısıdır de UNIX dosya sstem br dosya tanımlar. Br node , br node numarası çerr ve dosyanın dsk üzerndek konumunu, Unx dosya znlern de çeren özntelklerle ve dosyanın erşm zamanlarını tanımlar . Bu dosyanın node numarasını görüntüleyn : ls - / bn / ls

# Örnek 2



Dosya türü, dosyalardak bağlantılar çn de olablr (br ve aynı verlere, sabt bağlantılar çn br ve aynı node'a bağlı farklı dosya adları kullanılarak erşleblr ).

Ls programına sert br lnk oluşturun :

mkdr / bn / tmp ( hard lnk çn yen br dzn oluşturun ) ln / bn / ls / bn / tmp / ls

Şmd yen dosya adınızla lgl ayrıntıları görüntüleyn, / bn / tmp / ls : ls - l / bn / tmp / ls

# Örnek 3



Böylece, her k dosyanın da aynı node'u paylaştığını görüyoruz . Bu nedenle, bu dosyalardan brnn değştrlmes dğern de etkleyecektr.

İsmlerden brnn slnmes bağlantı sayacını azaltır. Sadece o zaman 0'a ulaşır node gerçekte kaldırılır: rm / bn / tmp / ls

# Örnek 3



Bağlantı / bn / çn ls yapışkan bt çn ayarlanmış olduğundan, ancak daha sonra bu bağlantıyı slemezsnz / tmp / dzn .

ls - ld / tmp /

İznlerde "t" y not edn ve yukarıda açıklanan anlama bakın.

Bağlantıyı kök olarak sleblrsnz:

sudo rm / bn / tmp / ls

# Örnek 5



/ Bn / tmp / after / bn / tmp / ls kaldırma şlemnde çerk görmüyoruz .

**5. İzinlerin sekizli gösterimi**

Stat komutu node'dan daha fazla blg görüntülemek çn kullanılablr : stat / bn / ls

# Örnek 6



File’s group

File’s owner

Unix file permissions

Bu blglere göz atın. Çıktının, erşm haklarını da çerdğn unutmayın.

le son kez dosya, erşlen değştrlmş ve ne zaman oldu node son değşt .

Stat'ten çıktı, blglern daha “nsan tarafından okunablr” br çıktıyla brlkte depolandığı formatı çerr.

Bldğmz gb, kullanıcı hesaplarına sstem tarafından UID'ler tarafından atıfta bulunulur , bu durumda UID

0'dır, çünkü dosya kök kullanıcıya attr. Benzer şeklde, gruplar da GID tarafından tanımlanır, bu durumda da 0'dır. Gerçek znler dört durumda (0-7 basamak), bu durumda “0755” olarak kaydedlr. Bu, (şmd tanıdık) nsan dostu çıktı olan “- rwxr - xr -x” anlamına gelr. Şmdlk lk sayı grubunu görmezden geleceğz, bu normalde 0, daha sonra özel anlamlarına döneceğz. Dğer üç oktet her br sadece çn kl temsl RWX her br br 0 ya da 1 olarak üç lk olarak temsl temsl eder, **u** ser, daha sonra **gr** RUBU, o zaman **o** Ther znn gerektrr. Dönüşüm yapmanın kolay ve hızlı br yolu sadece hatırlamaktır: ● r = 4

● w = 2 ● x = 1

Ve üç oktetn her brn üretmek çn bunları br araya getrn. Örneğn, rwx = kl 111 = (4 + 2 + 1) = 7. Aynı şeklde, rx = kl 101 = (4 + 1) = 5. Bu nedenle, “- rwxr - xr - x” = 755.

**6. Linux sistemindeki dosya izinlerini değiştirmek , kullanıcı, grup eklemek, dosya sahibini ve grubunu değiştirmek**

Şmd k kullanıcı arasında geçş yapmak çn br fırsata htyacımız olacak. Kullanıcıların lstes tarafından görüntüleneblr ked / etc / passwd gösteren / çerğn vs / passwd dosyasında.

# Örnek 7



Her kullanıcının / etc / passwd dosyasının okunableceğn görüyoruz . Şmd yen br kullanıcı, şfre öğrencs olan br öğrenc yaratalım. Süper kullanıcı haklarını sudo yardımı le kullanmak çn bu göreve htyacımız var . Sudo yapmasına zn verlen kullanıcıların hakları ( sudo grubundan) / etc / sudoers dosyasına dahl edld ked / etc / sudoers

# Örnek 8



Sudo grubu kullanıcılarının root le aynı znlere sahp olduğunu görüyoruz . Gruplar ve üyelernn edeblr nceled tarafından ked / etc / grup

# Örnek 9



Alex kullanıcısının sudo grubuna olduğu kadar, adm (sstem görevlern zleyeblr), cdrom ( CDROM'a erşeblr) ve dp ( çevrmel bağlantı çn ppp , dp vb. Araçları kullanablr ) gruplarına at olduğunu görüyoruz . Yen br kullanıcı tarafından yaratıldı sudo adduser student

# Örnek 10



Yen kullanıcı öğrencnn kullanıcı lstesnde olup olmadığını kontrol edn:

# Örnek 11



Grş dznnzde “ mysecret ” adlı br dosya oluşturun : ked > ~ / mysecret

# Örnek 12



Br dz çerk grn. “ Başkalarının görebleceğ ” br “sır” grdkten sonra Ctrl-D tuşlarına basın . İlk amacınız “ mysecret ” dosyanızın aynı sstemdek dğer kullanıcılar tarafından görülmemesn sağlamak . İlk önce yen oluşturduğunuz dosyanın znlern görüntüleyn:

ls –l ~ / mysecret Örnek 13



Oh hayır! O ' çok gzl değl ...

Chmod komutu br dosya znlern ayarlamak çn kullanılablr. chmod , mutlak sekzlk değerlere veya görecel değşklklere dayanarak znler ayarlayablr. Örneğn, octet'e dayalı br dosyadak znler ayarlamak çn chmod kullanablrsnz : 770 sahb ve grubuna rwx verr , ve dğerler zn alamaz. Örnek: chmod 770 / ev / tmp / bazı dosya

# Örnek 14



değştrlr . Veya görecel değşklkler yapablrsnz: ux sahb (kullanıcı) dosyayı yürütme olanağını kaldırır. Örnek: chmod ux / ana sayfa / tmp / somefle

# Örnek 15



Aynı şeklde, o + w dğernn dosyaya yazma yeteneğn eklerd Örnek: chmod o + w / ana sayfa / tmp / somefle

# Örnek 16



Mysecrets dosyanıza kendnze okuma ve yazma zn vermek çn chmod kullanın ve dosyaya zn vermeyen herkes: chmod 660 ~ / mysecrets

# Örnek 17



Dosyaya kullanıcı öğrencs adına erşmey deneyn: ked / ev / alex / mysecrets

# Örnek 18



Br dosya “~ / myshare ” oluşturun ve herkese okuma-yazma erşm vern. İznler doğru ayarlayıp ayarlamadığınızı test edn. Ayrıca “~ / myshare ” dosyasını bulmak çn dğer kullanıcılara gerekl znler vern .

# Örnek 19



“ Grubum paylaşımım ” oluşturun , yalnızca grubunuzdak herkese okuma erşm vern. İznler doğru ayarlayıp ayarlamadığınızı test edn.

# Örnek 20



Alex by grubuna kullanıcı öğrencs ekle s udo usermod –a –G alex öğrencs

# Örnek 21





Grubumu kullanarak yen br grup oluştur sudo groupadd benm grubum

# Örnek 22



Grubunu değştrn mygroupshare gelen alex çn mygroup kullanarak

Sudo chown : benm grubum ~ / benm grubum

Br dosyanın kullanım sahbn ve grubunu değştrmek çn chown new\_owner: newgroup dosyası

# Örnek 23



Ger değştrn mygroupshare grubun Alex ve vermek alex grubu sadece okuma erşm mygroupshare :

sudo chown : alex ~ / mygroupshare sudo chmod gw ~ / mygroupshare

# Örnek 24



“Çalışanlar” adında yen br grup oluşturun ve kendnz ve sınıf arkadaşınızla br dosya oluşturun.

( dğer kullanıcı) ortak çalışablr (her ks de düzenleneblr). İznler doğru ayarlayıp ayarlamadığınızı test edn. Her k kullanıcı da dosyayı düzenleyeblmel, ancak dğer kullanıcıların yazma erşmne sahp olmamalıdır.

# Örnek 25



# Mücadelesi 1. mkdr test

dokunma test / test1 test / test2 test / test3

Yen “test” dznndek tüm dosyaların znlern tekrar tekrar belrlemek çn tek br chmod komutu kullanın .

İpucu: “man chmod ”

**7. Komut umask**

Yen oluşturulan dosyamızın herkesn dosyayı okuyableceğ anlamına gelen znlerle başladığını unutmayın. Bu ayarı önleneblr **u** ser dosya oluşturma modu **maskes** ( umask ). Her şlem br umask'a sahptr : yen oluşturulan dosyaların znlern belrleyen br sekzl. Dosyalar çn varsayılan “666” ve yen yürütüleblr dosyalar çn “777” znlern kaldırarak çalışır (mantıksal br NOT'a dayanarak). Yan, br umask'ı “666” znlerne sahp yen dosyalar neden olacaktır “000” nn. Br Umask (varsayılan değer olan) “022” ve “644”, yan “veren RW - -r -r”. Umask sstem çağrısı ayarlamak çn kullanılablecek zn maskesn mevcut sürec. Geçerl umask değern kontrol edn :

Umask

# Örnek 26



# 2. meydan okuma.

Kullanılması Umask yerleşk komutu, kümenzle zn maskesn sadece yen dosyalar böylece rw szn tarafınızdan erşleblr ( ancak grubunuz veya başkaları çn değl ): umask XXX nerede XXX yendr umask'ı kullanmak.

Yen br dosya oluşturarak ve znlern kontrol ederek yen umask değernz test edn :

yen dosyaya dokun ls -l newflename

İznler “ rw -------” okuyor mu? Değlse, umask'ı değştrn ve tekrar deneyn.

**8. UID'yi (SUID) ayarlayın**

Bazen br kullanıcının her zaman sahp olmaması gereken znler gerektren şler yapablmes gerekr. Örneğn, passwd komutu şfrenz değştrmek çn kullanılır. / Etc / shadow dosyasına okuma ve yazma erşm gerekyor . Açıkçası her kullanıcı bu tür br erşme sahp olmamalı! Ayrıca, png komutu ham ağ erşmne htyaç duyuyor ... Yne her kullanıcının yapableceğ br şey değl. UNIX çözelts UID (SUID) olarak ayarlanır. SUID kullanarak, şlemlere başka br kullanıcı olarak çalıştırma zn verleblr. Örneğn, passwd'y çalıştırdığınızda , program aslında root olarak çalışır (çoğu Unx sstemlernde). Aslında, her şlem aslında aşağıdakler dahl brden fazla kmlğe sahptr:

* Gerçek UID (rud): Kullanıcı komutunu çalıştıran km
* etkl UID (EUID): yönlü br süreç şlenr

Etkl UID'nn nasıl belrtldğne br bakın: ls - l / usr / bn / passwd

# Örnek 27



Dosya znlerndek "s", UID dosyasının etkn UID olarak kullanılacağı anlamına gelr. Çalıştırmak stat / usr / bn / passwd

Sonra başka br kullanıcıya geçn (bob) ve passwd komutunu çalıştırın

# Örnek 28



node çndek lk (kullanıcı) zn octetnde saklanır . Başka br bash sekmesnden çalışan şlemler görüntüleme le ps – af

# Örnek 29



bz görüyoruz passwd rağmen UID = köküyle çalışır bob tarafından başlatılan arasında. SUID le tüm programları bul / home / by sudo fnd / home –perm -4000 –tp f – z ve stat kullanarak bulguları kontrol edn:

# Örnek 30



Program erşm krterler aşağıda açıklanmıştır.

**9. C de bir SUID programı yazmak**

İçerğnze erşm vermek çn br SUID programı oluşturacaksınız.

“ Mysecret ” dosyasını çalıştıran herkese, dosyaya doğrudan erşm paylaşmadan .

“~ / Mysecrets ” uygulamasına yalnızca kullanıcı tarafından erşlebldğnden emn olun : ls -la “ rw ------- ” çn bu dosyanın.

# Örnek 31



“ Accessmysecret.c ” adlı yen br dosya oluşturarak br C programı oluşturun :

v accessmysecret.c

Unutma, v kalıcı. Ekleme moduna grmek çn “” ye basınız, daha sonra şu kodu grnz:

#nclude < stdo.h >

#nclude < stdlb.h >

#nclude <sys / types.h >

#nclude < unstd.h > #nclude < errno.h > nt man () {

prntf ( "UID GID \ n" "Gerçek% d Gerçek% d \ n" "Etkl% d Etkl% d \ n", getud (), getgd () , geteud ( ), getegd ()) ;

DOSYA \* fp = fopen ( " mysecrets ", "r");

f ( fp == NULL) {

prntf ( "Hata: Dosya açılamadı"); çıkış ( EXIT\_FAILURE);

}

char c;

whle ((c = getc ( fp ))! = EOF) {

putchar ( c);

}

putchar ( '\ n');

ger EXIT\_SUCCESS;

}

# Örnek 32



Değşklklernz kaydedn ve çıkın (Esc, “ : wq ”). Herhang br başka metn edtörünü kullanablrsnz





Programı derleyn (çalıştırılablr br dosya oluşturmak çn C kodunu kullanır):

gcc accessmysecrets.c -o Instagram Hesabındak Resm ve Vdeoları accessmysecrets

Dosyanın znlern ( chmod kullanarak ) setud olarak ayarlayın :

chmod u + s Instagram Hesabındak Resm ve Vdeoları accessmysecrets SUID çeren znler kontrol et: ls -l accessmysecrets

# Örnek 33



Programı çalıştır:

./ Instagram Hesabındak Resm ve Vdeoları accessmysecrets

# Örnek 34



Başka br kullanıcıya geçn ve programı yürütün: / home / kullanıcı adınız / accessmysecrets

# Örnek 35



çndekler at mysecrets sze sırlarını dosyaya erşm olmayan ble, dosya doğrudan .

# Meydan 3

S başka br kullanıcıya cadı ve SUID kullanmak accessmysecrets programı olsun sahbdr kullanıcının dosyalarının herhang brne okuma erşm!

İpucu: Bu kodda br güvenlk sorunu var.

Başka br pucu: sert bağlantılar hakkında düşünün.

**Çözüm:**

Açarken mutlak dosya adı kullanılmamasının neden olduğu br güvenlk sorunu var dosya, bu “açılır mysecrets yerne” / home / user / den “ mysecrets ”. Unutma, herhang br kullanıcı br dosyaya sert br bağlantı oluşturablr (bu nedenle SUID’nn br “kopyasını” yapablr. stedkler yerde program yaparlar).

SUID programına, saldırganın yazableceğ br dznde bağlantı sağlayın,

Daha sonra , SUID kullanıcısının sahp olduğu herhang br dosyaya da sert br bağlantı yapın ve " mysecrets " olarak adlandırın.

Programı çalıştırdığınızda programın aynı dzn, o zaman yazacak Dosyanın çerğn çıkarın .

Bu güvenlk açığından aşağıdak şeklde yararlanablrsnz: su - öğrenc ln / home / user / accessmysecrets / tmp / erşm ln / home / user / başka br dosya / tmp / mysecrets / tmp / erşm

# 4. zorluk

Yukarıdak güvenlk açığını gdermek çn programı değştrn.

# Meydan 5

Programı, mysecrets dosyasının yalnızca lk satırı olacak şeklde değştrn. başkalarına gösterlr .

# Meydan 6

Programı, komut dosyası UID'y denetleyecek ve yalnızca devam edecek şeklde değştrn çn belrl br kullanıcı (örneğn, kullanıcı kökü se).

İpucu: “adam getud

**10. Linux Genişletilmiş ACL'ler**

Standart Unx znlern araştırdık . Modern Lnux sstemler (ve dğer bazı Unx tabanlı sstemler) artık daha eksksz (ve karmaşık) ACL desteğne sahptr. Daha önce belrtldğ gb, br erşm kontrol lstes ( ACL), br nesnenn (kaynak) bağlanmış ve erşm tür le brlkte, erşm zn verlen tüm konularda (kullanıcılar / aktf kşler) lstelyorsa yetkl .

Senn üzernde br dosya ACL ayarlayın mysecrets kullanarak, dosyanın setfacl komutu: setfacl -mu: öğrenc: r ~ / mysecrets

# Örnek 36



Stat programının genellkle ACL farkında olmadığını, bu nedenle dışında hçbr şey bldrmeyeceğn unutmayın. Her zamank:

stat ~ / mysecrets

Ls programı Dosya ACL'lern algılamak çn kullanılablr: ls -la ~ / mysecrets

- rw - r ----- + 1 clffe kullanıcıları 22 Şub 28 11:47 mysecrets

Çıktının br “+” çerdğn unutmayın. Bu br ACL'nn yernde olduğunu gösterr. İznler görüntülemek çn getfacl kullanın :

getfacl ~ / mysecrets

Br veya daha fazla belrl kullanıcıya (dğer sınıf üyeler) mysecrets dosyanıza okuma erşm vermek çn Lnux Fle ACL'ler kullanın .

ACL'ler kullanarak, herhang br gruba (seçtğnz) grubum paylaşım dosyama okuma-yazma erşm vern . Yen ekledğnz grup znn kaldırın.

Örnek: setfacl -xg : staff fle

**11. Sonuç**

Bu noktada sahp olduğunuz:

1. Dosya znler, hard lnkler ve node'lar hakkında blg ednld
2. İznlern sekzl göstermler hakkında blg ednld
3. Chmod kullanarak belrl kullanıcılara ve gruplara erşm vermek çn değştrlen Unx dosya znler
4. Yen dosyalara uygulanan znler değştrmek çn umask kullanıldı
5. Set UID (SUID) hakkında daha fazla blg sahb olmak, C hakkında daha fazla blg sahb olmak ve br SUID C programı derlemek
6. Ayrıca, kendnz çn braz daha programlama yapmış olablrsnz.
7. Daha gelşmş güvenlk poltkaları yapılandırmak çn Lnux Extended ACL'ler kullanıldı Afern!

**Referanslar**

1. Z. Clffe Schreuders . Erşm kontroller ve Lnux / Unx dosya znler, z.clffe.schreuders.org/ edu /ADS/Access%20Controls.pdf

1