

Berger (Tek Yönlü Bit Hata Sezme) Kodu Ekleme

Berger kodu ile tek yönlü bit hatalarını sezme için kullanılabilecek aşağıda verilen assembly dili kod parçasının, 1. parametre ile verilen giriş dosyasını açtıktan sonra 2. parametre ile verilen çıkış dosyasına tüm karakterleri kopyalayan **arg_tasm.asm** adlı assembly program dosyasındaki **CopyLoop** döngüsü içine yerleştirilmesi yeterlidir. Bu kod, okuma tamponundaki (başlıkta dosya türünü belirten **Sihirli Sayı** için ayrılan ilk iki byte dışındaki) tüm karakterlerin Berger kodunu, unicode karakterlerin kullanılmayan anlamlı kısmı içine yerleştirmektedir.

```
lea     esi, ReadBuffer      ; KaynakIndisi esi <- OkumaTamponu
lea     edi, WriteBuffer     ; HedefIndisi edi <- YazmaTamponu
mov     al, HdrPrscd         ; Baslik aktarildi mi?
cmp     al, 1
je      HdrPrc               ; aktarilmissa tum tamponu kopyalamaya git
mov     al, [esi]             ; OkumaTamponundan al
mov     [edi], al            ; YazmaTamponuna aktar
inc     esi                  ; Kaynak ve Hedef indislerini ilerlet
inc     edi
mov     al, [esi]             ; OkumaTamponundan al
mov     [edi], al            ; YazmaTamponuna aktar
inc     esi                  ; Kaynak ve Hedef indislerini ilerlet
inc     edi
mov     al, 01h              ; Baslik aktarildi
mov     HdrPrscd, al         ; Baslik aktarildi bilgisini guncelle
mov     ecx, BUFFER_SIZE - 2 ; Baslik haric geri kalan tampon boyu
jmp     Trnsfr               ; Aktarmaya git

HdrPrc:
mov     ecx, BUFFER_SIZE     ; Baslik aktarildigi icin tum tampon boyu

Trnsfr:
mov     al, [esi]             ; Karakteri YazmaTamponuna aktar
mov     [edi], al
mov     bl, 0                ; Birler sayacini sifirla

CountOnes:
cmp     al, 0                ; 1 kalmadi ise
je      Done                 ; sayma sonlanir
shr     al, 1                ; saga 1 bit (elde bayragina) kaydir
jnc     Skip                 ; elde bayragi sifirsa 1 sayisini degistirmez
inc     bl                   ; elde bayragi 1 ise sayac artar

Skip:
jmp     CountOnes            ; 1 saymaya devam et

Done:
inc     esi                  ; Unicode da kullanilmayan karakteri gosterir
inc     esi                  ; Sonraki gecerli karakteri gosterir
inc     edi                  ; Berger kodu yazilacak yeri gosterir
not     bl                   ; 1 sayisinin tumleyeni
mov     [edi], bl            ; Berger kodunu YazmaTamponuna aktar
inc     edi                  ; Sonraki gecerli karakteri gosterir
dec     ecx                  ; Tampon aktarimi bitene kadar don
jnz     Trnsfr
```

Dosya başındaki başlık bilgisinin yalnız bir defa işlenmesi için başlangıç değeri bilinen veri tanımlamalarına eklenen;

```
HdrPrscd db 0
```

adlı global değişken kullanılmış ve başlangıçta henüz başlığın işlenmediği anlamına gelen 0 değeri atanmıştır. Başlık içindeki **Sihirli Sayı**'yı tutan ilk 2 byte, çıkış tampouna değiştirilmeden aktarıldıktan sonra bu değişken;

```
mov al, 01h
mov HdrPrscd, al
```

emirleri ile başlığın aktarıldığını global değişkene saklamakta ve bundan sonraki tüm aktarımların tampon içindeki tüm karakterleri içermesi gerektiğini bildirmektedir.

Dosya kopyalayan **arg_tasm.asm** içine, Berger kodu hesaplayan kod parçasının eklenmesi sonucunda oluşan **BergerKodla.asm** adlı programı, derledikten sonra çalıştırılabilir dosyayı **td32.exe** ile açıp giriş ve çıkış dosya adlarını **Run** menüsü altındaki **Arguments** seçeneği ile vererek çalıştırırsanız, giriş dosyası Sihirli Numarasını koruyarak unicode karakterlerin kullanılmayan anlamlı kısımlarına Berger kodu yerleştirilmiş çıkış dosyasını oluşturduğunu görebilirsiniz.

Büyük harfleri içeren unicode **giris.txt** dosya içeriğini, Linux işletim sisteminde dosya içeriğini herhangi bir sayı tabanında görüntüleyebilen Octal Dump (od) komutu ile;

```
od -A x -t x1z -v giris.txt
000000 ff fe 41 00 42 00 43 00 44 00 45 00 46 00 47 00 >. .A.B.C.D.E.F.G.<
000010 48 00 49 00 4a 00 4b 00 4c 00 4d 00 4e 00 4f 00 >H.I.J.K.L.M.N.O.<
000020 50 00 51 00 52 00 53 00 54 00 55 00 56 00 57 00 >P.Q.R.S.T.U.V.W.<
000030 58 00 59 00 5a 00 >X.Y.Z.<
```

olarak görebilirsiniz. **BergerKodla.exe** programı ile bu dosyadaki karakterlere Berger kodu eklendiğinde üretilen **cikis.txt** dosyasında da;

```
000000 ff fe 41 fd 42 fd 43 fc 44 fd 45 fc 46 fc 47 fb >. .A.B.C.D.E.F.G.<
000010 48 fd 49 fc 4a fc 4b fb 4c fc 4d fb 4e fb 4f fa >H.I.J.K.L.M.N.O.<
000020 50 fd 51 fc 52 fc 53 fb 54 fc 55 fb 56 fb 57 fa >P.Q.R.S.T.U.V.W.<
000030 58 fc 59 fb 5a fb >X.Y.Z.<
```

karakter Berger kodlarının doğru hesaplanarak unicode karakterlerin kullanılmayan (gri arkaplan ile belirtilen) üst kısımlarına yerleştirildiği görülmektedir.

Benzer şekilde, Berger kodu ile tek yönlü bit hatalarına karşı korunmuş dosyadan okuduğu karakterlerin Berger kodunu denetleyerek bit hatası içeren karakterleri # sembolü ile işaretledikten sonra çıkış dosyasına yazan Assembly dili kodunu oluşturarak test edebilirsiniz.

Not: Komut satırından verilen giriş ve çıkış dosya adlarını alarak, giriş dosyasındaki tüm karakterlerin Berger kodunu hesaplayan ve unicode karakterlerin kullanılmayan anlamlı kısmına yerleştiren **BergerKodla.asm** [dosyası](#) ders sayfasına koyulmuştur.

24.05.2016