

Merhabalar,

Cybercamp19 kamp son ctf yarışmasında çıkan tersine mühendislik sorularının yapmış olduğum çözümleri şu şekilde;

Program mekaniklerine değinmek amacıyla ilk soru için detaylı anlatımdır.

Rev50.exe

Çalıştırılabilir dosyamızı yürüttüğümüzde bize sorduğu soruya herhangi bir cevapla karşılık verince BAD! Tuzağına düşüyoruz.

```
Administrator: C:\Windows\system32\cmd.exe
C:\Users\IEUser\Desktop>rev50.exe
Gizli mesaj ne ? : test
Bad!
```

Herhangi bir exe disassemble edebilen bir programla baktığımızda (ben x86WinDbg kullanmak istedim) , programın giriş noktası (Entry point'e) ulaşıyoruz.;main fonksiyonumuz. Bu noktada ki kontrolleri anlamak için öncelikle verilen mesajları bularak başlıyorum. Main fonksiyonun frame'ı içerisinde kullanılan stringleri listelemek istiyorum . Eğer stringler bir şekilde saklanmadıysa işimiz oldukça kolay.

Address	Disassembly	String
0040128A	mov dword ptr ss:[esp],rev50.403000	"Gizli mesaj ne ? : "
004012CD	mov dword ptr ss:[esp],rev50.403014	"ss"
004012DD	cmp eax,rev50.403018	"IZWGCZ33IJ2UIYLMFBGC43MMFXGO2LDPU===="
004012E4	mov dword ptr ss:[esp],rev50.403041	"Good!"
004012F2	mov dword ptr ss:[esp],rev50.403047	"Bad!"

Program içerisindeki stringleri bulmak için genellikle "search for > all referenced strings" şeklinde seçeneklerden ulaşabilirsiniz.

Öncelikle "Bad!" mesajını hangi kontrol noktası kullanmış buna bir bakalım: Yine kolay bir şekilde Bad! String'ine tıkladığınız zaman sizi kod akışı içerisinde çağırdığı fonksiyona götürecektir.

call <JMP.&scanf>	
movsx eax,byte ptr ss:[ebp-1]	
cmp eax,rev50.403018	403018: "IZWGCZ33IJ2UIYLMFBGC43MMFXGO2LDPU===="
jne rev50.4012F2	
mov dword ptr ss:[esp],rev50.403041	403041: "Good!"
call <JMP.&puts>	
jmp rev50.4012FE	
mov dword ptr ss:[esp],rev50.403047	403047: "Bad!"

Resimde ki akışa bizi götürdü ve fonksiyona genel olarak baktığımız da bizden scanf() fonksiyonu ile aldığı girdi değerinin ilk karakterini ebp-1 (tüm girdinin tutulduğu stack) 'ten BYTE PTR ile sadece bizim girdimizin sadece ilk karakterini alıp IZWGC.. stringini gösteren data adresi ile karşılaştırıyor. Bu işleme göre programa patch yapmadan asla GOOD mesajına ulaşma şansımız olmayacak. Zaten program bizden onu da istemiyor. Data adresinde bulunan stringi BASE32 decode edince elde ettiğimiz değer puan almak için yeterli olacak.

IZWGCZ33IJ2UIYLMFBGC43MMFXGO2LDPU

Flag{BuDahaBaslangic}