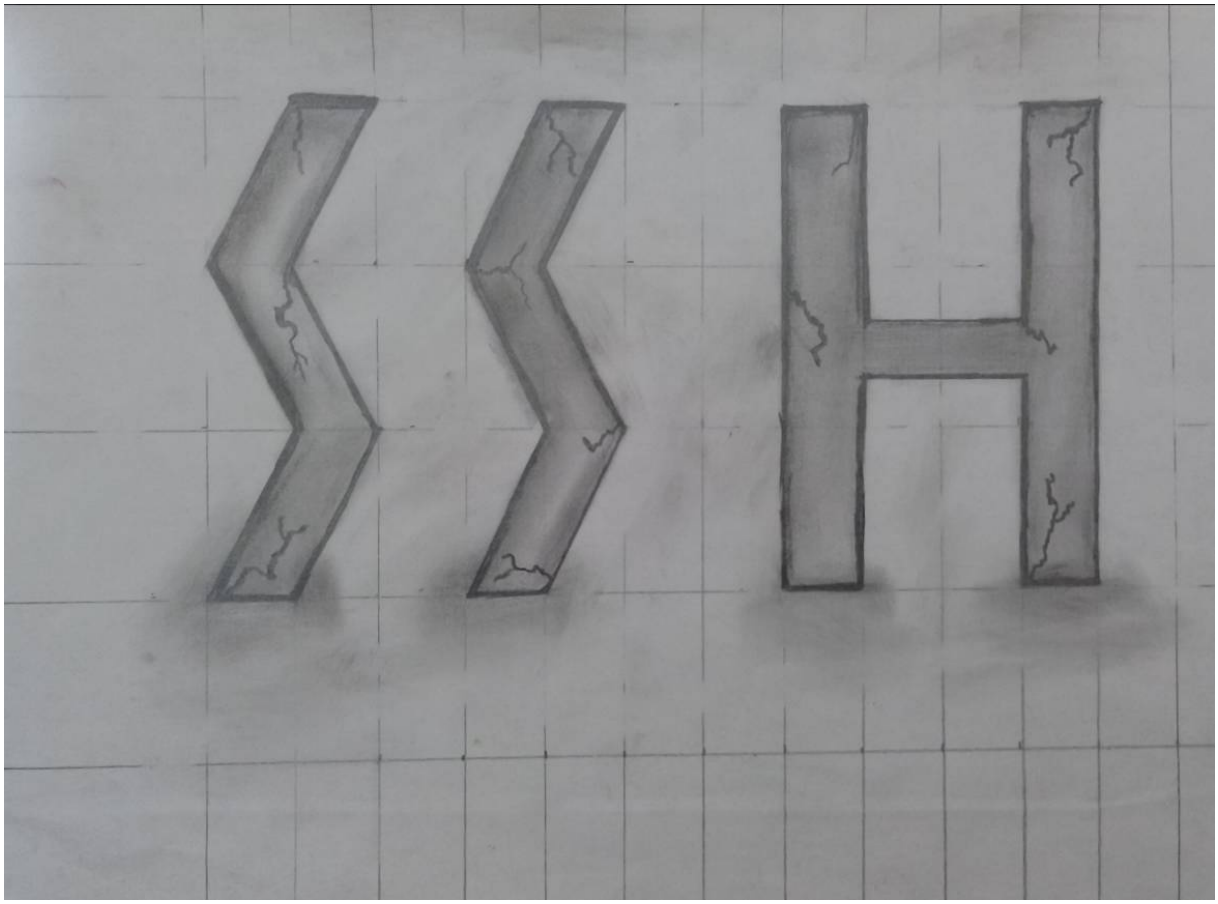


SSH Client Backd00r

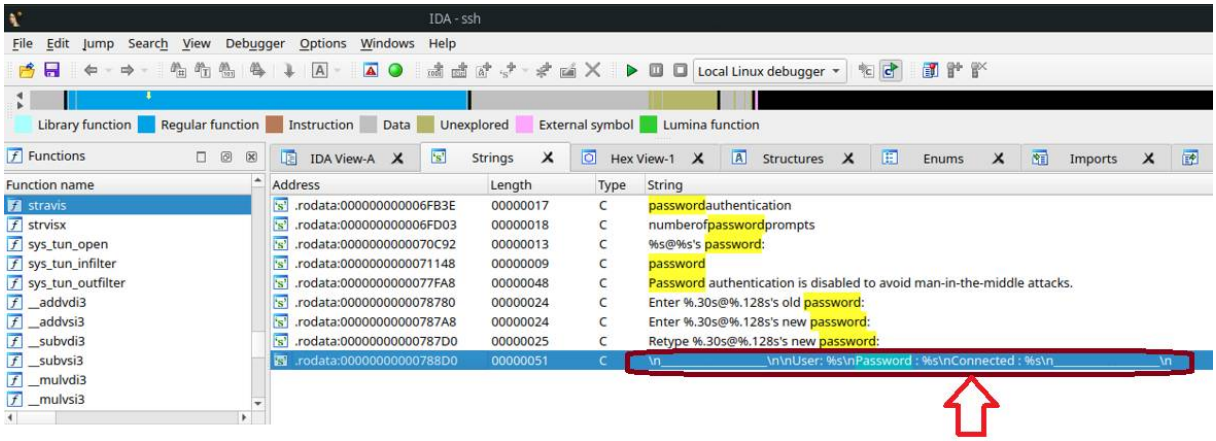
Tacettin Karadeniz
tacettinkaradeniz {at} proton.me

Geçmişten günümüze sistemlere yetkisiz erişim için kullanılan taktikler değişmektedir. Sistemde zafiyet tespit edildiğinde sisteme erişim sağlayan yetkisiz kişi ya da kişilerin yapacağı ilk işlemlerden biri de sistemde arka kapı (Backdoor) oluşturmaktır. Sistemde oluşturulan arka kapıların çoğunluğu; fark edilmeden kullanıcı oluşturma, harici erişim için port açma, kullanılan sunucu tabanlı yazılımlarına müdahale edilerek istenildiği zaman sunucuya bağlanma yöntemleridir.
Bu tür aksiyonların haricinde uygulamalarda (sunucu özelliği harici) yapılan değişikliklerle sistemde bir arka kapı etkisi oluşturulur.

SSH istemci uygulamasının kaynak kodunda yapılan değişiklikler ile sisteme yüklenmesi sonucunda sistem nasıl tehlikeye düşüyor, bunu irdeleyeceğiz.
SSH{Secure Shell}, sistemler arasında diğer tabirle uzak sunuculara(remote control) ve bilgisayarlara bağlanmak maksadıyla kullanılan bir araç/ağ protokolüdür. Güvenli bir şekilde dosya transfer işlemi içinde kullanılır. SSH ağ protokolü standart port 22/TCP olarak kullanılmakta. SSH istemci(client) uygulaması ile SSH sunucusuna bağlantı işlemi gerçekleşir.

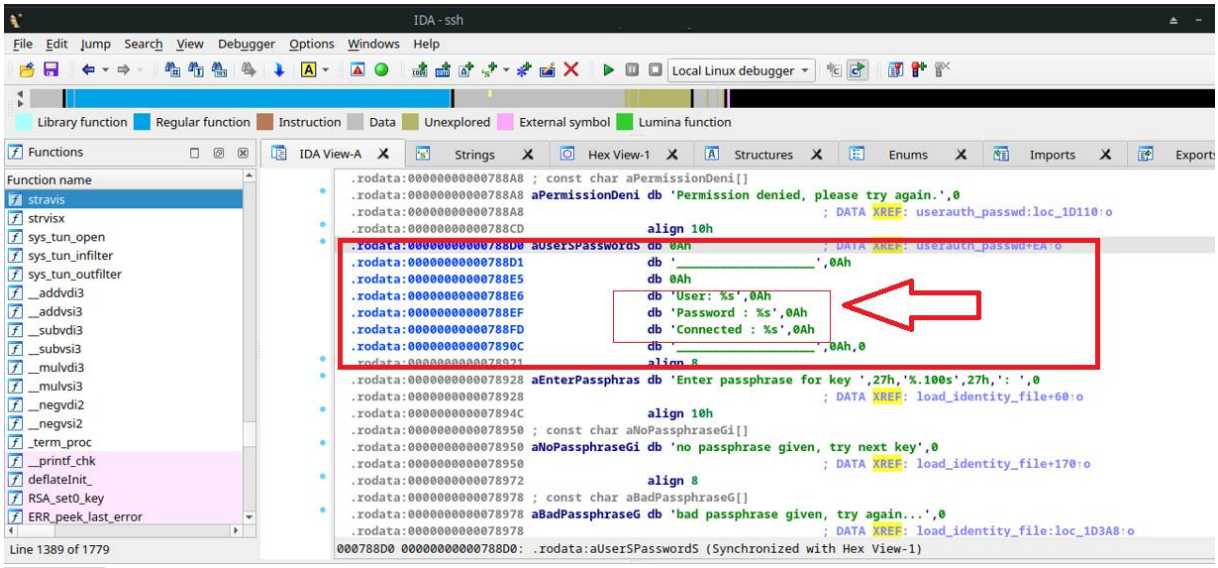
Yukarıda kısaca bahsettiğim SSH istemcisinde yapılan değişiklikle sunucuyu nasıl tehlikeli bir duruma getirir?
Çok kullanıcılı bir sunucu (server/örn: Seedbox server) yetkisiz erişim ile (#)root(#) yetkisi alan bir saldırgan sistemde tespit edilse dahi tespit öncesi diğer kullanıcıların şifrelerini ele geçirmek isteyecektir. Bu yollardan bazıları Sniff yöntemi (Network Sniffing), *nix sistemde /etc/shadow dosyasında ki bilgiler ışığında(yetkisi kısıtlı kullanıcı erişemez) kullanıcı şifrelerinin kırılmasıdır(John The Ripper, Hashcat).
Saldırgan, sistemde root(#) yetkisine eriştiğinde SSH client uygulamasındaki değişiklik ile sunucudaki diğer kullanıcıların ssh komutunu kullanmasıyla başka sunuculara(iç ağdaki başka sunucu olabilir) bağlanmasıyla zorlanmadan şifreleri ele geçirebilir.

IDA(The Interactive Disassembler) ile `ssh` istemci uygulamasındaki strings(Resim 1) içerikleri incelendiğinde (**User: \ Password: \ Connected:**) söz dizimleri dikkat çekiyor.

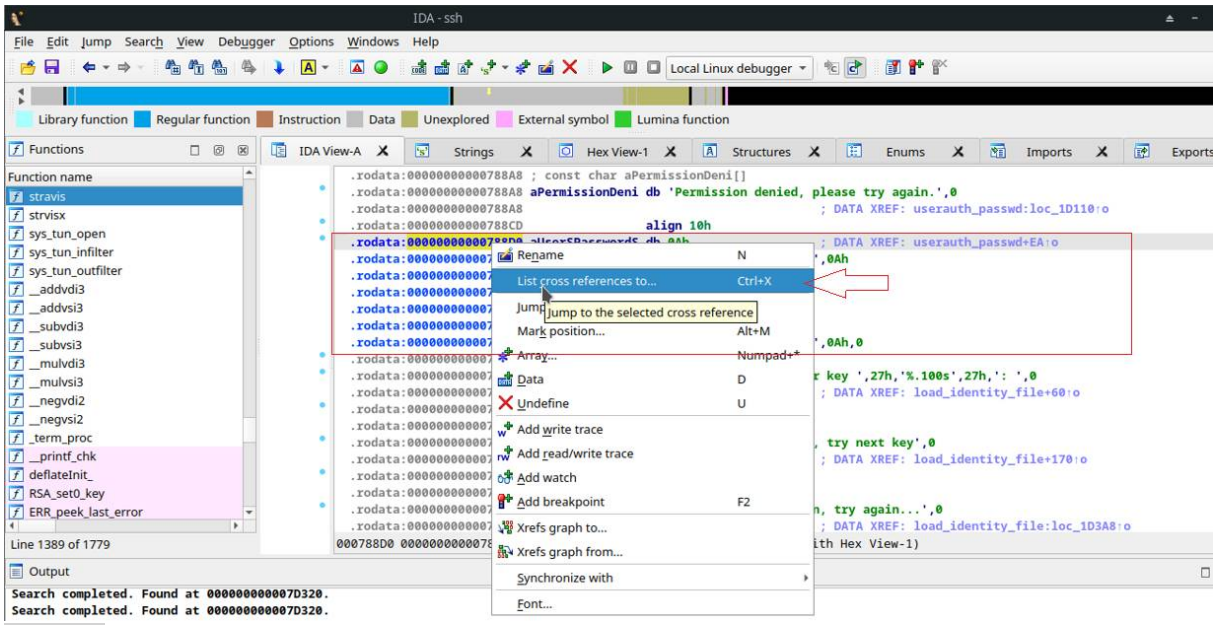


(Resim 1)

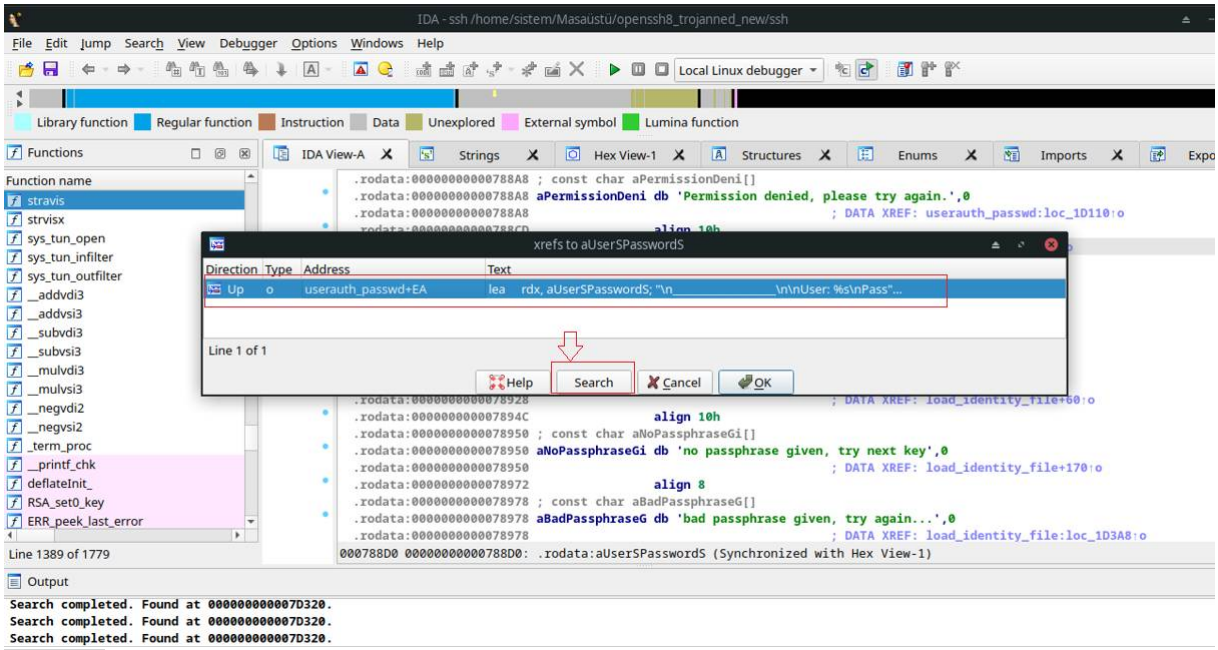
Dikkat çeken bu ifadelerin neyi ifade ettiğini bulmak için durumlarını IDA ile incelemeye devam edelim. Resim 2, Resim 3, Resim 4 ve Resim 5 de yer alan ssh client uygulaması için çalışma sürecinde normal olmayan durumlar görülüyor.



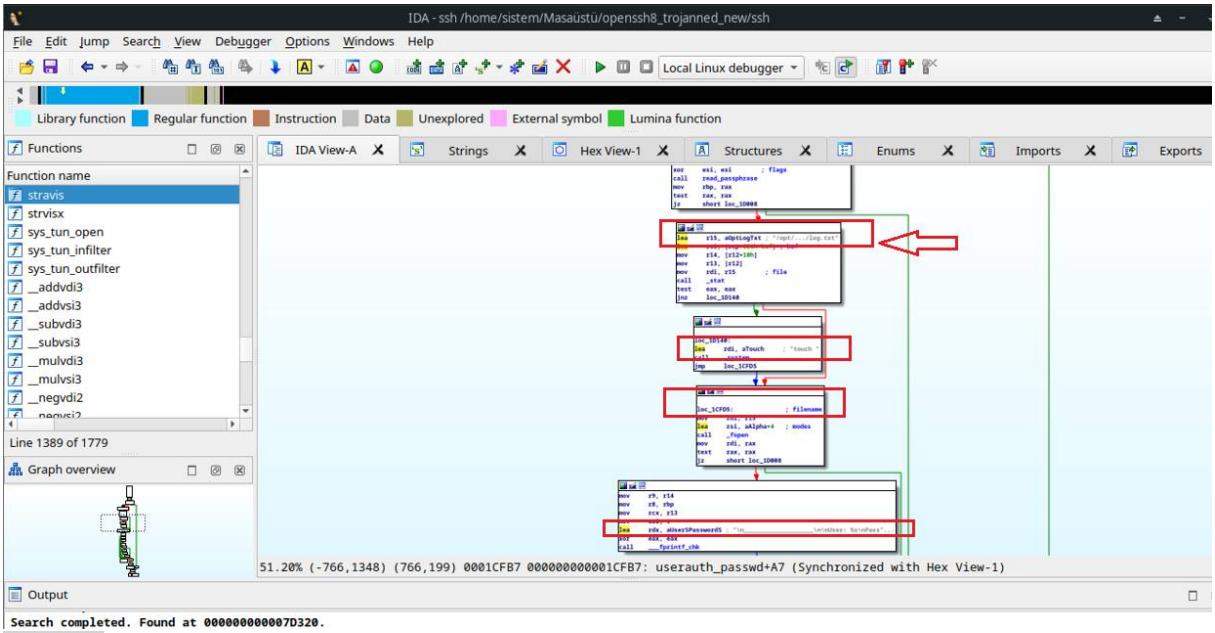
(Resim 2)



(Resim 3)



(Resim 4)



(Resim 5)

Resim 5 de yer alan uygulama diyagramında "**/opt/…/log.txt**" ve "**touch** " ifadeleri ssh komutu için şüphe uyandırıcıdır. Uygulama sistem üzerinde **log.txt** dosyasına erişim sağlamakta ve log.txt dosyasının olup olmadığını kontrol ediyor, bu log.txt dosyası yok ise `touch` komutunu devreye alıyor Konsol ortamında ssh komutunu çalıştırıp süreci izleyelim.

ssh ile iç ağdaki bir bilgisayar bağlantıyı sağladığımızda herhangi bir aksilik görünmüyor. Normal bir bağlantı gerçekleşti.

[sistem@sys]$ ssh -l domain5 192.168.5.9
domain5@192.168.5.9's password:

┌──(domain5㉿yerel5)-[~]

└─$

ssh uygulamasının sürecini konsol üzerinden takip etmeye devam edelim. "**ps**" komutu ile "**ssh**" süreç numarasını(PID) bulalım.

[sistem@sys]$ ps -ef | grep ssh

sistem 126779 105387 0 15:31 pts/2 00:00:00 ssh -l domain5 192.168.5.9

/proc dizininden çalışan uygulamaların bilgilerine ulaşabiliriz. ssh süreç numarası **126779** olarak gördük. Bulduğumuz süreç numarası üzerinden /proc dizini altında ssh için durum analizini gerçekleştirelim.

[sistem@sys]$ ls -la /proc/126779/fd/

dr-x------ 2 sistem sistem 8 Nis 19 15:31 .
dr-xr-xr-x 9 sistem sistem 0 Nis 19 15:31 ..
lrwx------ 1 sistem sistem 64 Nis 19 15:31 0 -> /dev/pts/2
l-wx------ 1 sistem sistem 64 Nis 19 15:31 1 -> /dev/null
lrwx------ 1 sistem sistem 64 Nis 19 15:31 2 -> /dev/pts/2
lrwx------ 1 sistem sistem 64 Nis 19 15:31 3 -> 'socket:[220577]'
l-wx------ 1 sistem sistem 64 Nis 19 15:31 4 -> /opt/.../log.txt
lrwx------ 1 sistem sistem 64 Nis 19 15:31 5 -> /dev/pts/2
lrwx------ 1 sistem sistem 64 Nis 19 15:31 6 -> /dev/pts/2
lrwx------ 1 sistem sistem 64 Nis 19 15:31 7 -> /dev/pts/

"**l-wx------ 1 sistem sistem 64 Nis 19 15:31 4 -> /opt/.../log.txt**" olan satıra dikkat ettiğimizde ssh uygulamasının **/opt/…/** dizini altında **log.txt** dosyasına bir erişim gerçekleştirdiğini görüyoruz.
ssh istemcisi için normal olmayan, ilginç bir durum. IDA ile analiz ettiğimizde aynı durumu görmüştük (Resim 5).

ssh istemcisinin /opt/…/ dizini içerisinde eriştiği log.txt dosyasındaki sır nedir?

[sistem@sys]$ cat /opt/.../log.txt

_____

User: domain5
Password : karaelmas
Connected : 192.168.5.9

_____

Görüyoruz ki log.txt dosyasında ssh istemci ile bağlanılan IP adres ve kullanıcı bilgisi kaydedilmiş.

Saldırgan, sistem ssh istemcisini değiştirerek sistem üzerinden ssh ile bağlanılan uzak sunucu IP adresi, kullanıcı ve şifre bilgilerinin kaydetmesini sağlamış. Yeter ki ssh komutunu kullansın.

Ayrıca tespit için YARA kuralı:

```
import "elf"

rule ssh_client_backdoor

{
   meta:
      author = "Tacettin Karadeniz"
      date = "2024-04-17"
      description = "SSH Client Backd00r"
   strings:
      $str0 = "touch /opt/.../log.txt"
   condition:
      uint16(0) == 0x457f and
      filesize < 3MB and
      all of them
}
```

[sistem@sys]$ yara -r ara.yar ./test/ssh

ssh_client_backdoor ./test/ssh

Kaynaklar

- *{IDA}*
https://hex-rays.com/ida-free/

- *{OpenSSH 8 Password Backdoor}*
https://packetstormsecurity.com/files/178064/OpenSSH-8-Password-Backdoor.html

-*{YARA}*
https://github.com/virustotal/yara

- *{Gizliliğin Anatomisi}*
https://www.exploit-db.com/docs/38893