ÖN BİLGİ

Bu belgenin resmi adresi bulunamamıştır. Alternatif adreste yedeklenmiştir. Bu belge

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/G%C3%BCvenli%20Kanallardan%20%C4%B0leti%C5%9Fim%20-%20SSH.pdf

adresindeki makaleye çalışılarak elde edilen notlarımı kapsamaktadır. Bu çıkarılan notlar belgemde alıntılar ve/veya kişisel ilavelerim mevcuttur.

1)

SSH Nedir?
SSH (Secure Shell) ağ üzerinden başka bilgisayarlara erişim sağlamak, uzak bir bilgisayarda komutlar çalıştırmak ve bir bilgisayardan diğerine dosya transferi yapmak amacıyla geliştirilmiş bir protokoldür. Aynı işi telnet de yapmaktadır, fakat ssh veri alışverişini şifreleyerek hatlardan iletmektedir. Bir farkı budur. SSH güvensiz kanallar (internet) üzerinden güvenli haberleşme olanağı sağlar. Bir iletişimde SSH aşağıdaki belirtilen temel unsurları temin eder:

- Authentication (Yetkilendirme)
- Encryption (Şifreleme)
- Integrity (Bütünlük)

(page 1)

2)

SSH Versiyonları

- SSH1 //Tatu Yiönen tarafından geliştirilen ilk SSH Ürünü
- SSH2 //Tatu Yiönen tarafından geliştirilen ikinci SSH Ürünü
- SSH-1 // Protocol 1
- SSH-2 // Protocol 2

Günümüzde yaygın kullanımda olan ve kullanımı tavsiye edilen ssh sürümü SSH-2 'dur.

(page 1)

3)

OpenSSH Tarihçesi

1995 yılında Tatu Yiönen tarafından geliştirilen SSH1 bir süre sonra geliştiricisi tarafından kaynak kodlarıyla berabet özgür olarak dağıtılmaya başlanmıştır. Bu kaynak kodlardan ilhamla SSH protokolünün özgür bir uyarlaması olan OpenSSH doğmuştur. OpenSSH SSH versiyonlarından son özgür nitelikte olan ssh1.2.12 versiyonundan türetilmiştir.

(page 1)

4)

Göreceli olarak özgür yazılım projeleri arasında OpenSSH en fazla kullanılan yazılımlardan biri olmuştur. İnternette kullanılan SSH sunucularının büyük çoğunluğu (yaklaşık %90'ı) OpenSSH kullanmaktadır.

(page 2)

5)

OpenSSH'ın birçok platforma uyarlanmış versiyonları mevcuttur. Örneğin linux ve windows bunların arasındadır.

(Page 2)

6)

SSH Kullanım Alanları

SSH güvenli iletişimin gerektiği her ortamda kullanılabilir niteliktedir. Mesela POP3 servisi ağ üzerinden tüm iletişimini şifrelenmemiş şekilde gerçekleştirir. Yani karşı taraftan alınan mail'in içeriği hat üzerinden plain text formatında bize gelir. Eğer pop3 servisini SSH'la buluşturursak gelen mail'lerin hat üzerinden şifrelenmiş olarak bize intikal etmesini sağlayabiliriz.

Page 2)

7)

OpenSSH birçok Unix/Linux dağıtımı ile öntanımlı olarak gelmektedir.

> ssh

Output:

```
usage: ssh [-1246AaCfgKkMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
           [-D [bind_address:]port] [-E log_file] [-e escape_char]
           [-F configfile] [-I pkcs11] [-i identity_file]
           [-L [bind_address:]port:host:hostport] [-l login_name] [-m mac_spec]
           [-O ctl_cmd] [-o option] [-p port]
           [-Q cipher | cipher-auth | mac | kex | key]
           [-R [bind_address:]port:host:hostport] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] [user@]hostname [command]
```

(Page 3)

8)

Temel SSH Kullanımı

SSH Client olarak herhangi bir SSH Server'a bağlanıldığında öncelikle client'ın konsol ekranına SSH Server'ın kimlik bilgisini temsil eden RSA anahtarı gelir Ardından bu server'a hiç bağlanmadığı ve cache'e ekleneceği uyarısı gelir (Eğer sunucu IP adresinde ya da SSH sunucusunun kimliğinde bir değişiklik olursa bu uyarı farklı şekilde ekrana gelir). Uyarının akabinde ise bağlanılan SSH sunucusunun belirtilen kullanıcısının şifresinin girilmesi beklenilir:

> ssh sshSunucuIP

Output:

The authenticity of host 'ssh_sunucu (14.2.7.x)' can't be established.
RSA key fingerprint is f3:ce:14:99:d7:19:44:ca:ff:5e:83:b6:79:52:4e:45.
Are you sure you want to continue connecting (yes/no)?yes
Warning: Permanently added 'ssh_sunucu,14.2.7.x' (RSA) to the list of known hosts.
huzeyfe@SSH_SUNUCU_IP's password:

Burada bir noktaya dikkat edin. SSH Sunucusuna bağlanırken bize kendi kabul ettiği varsayılan kullanıcıyı dikta etmiştir ve onun şifresini girmemizi istemiştir. Bu varsayılan kullanıcının adı hedef makinanın yerel yetkili kullanıcısının adını teşkil eder. Bu kullanıcı harici bir kullanıcıya bağlanmak istersek -l parametresini kullanabiliriz:

> ssh -l rapsodi sshSunucuIP

veya

> ssh rapsodi@sshSunucuIP

Yukarıdaki iki kullanımda aynı işlemi yerine getirir.

NOT: Hedef makinada şifresi oturum açma imkanı sunan ve SSH Server'ın kurulumunda varsayılan olarak gelen anonymous kullanıcısı mevcut mu diye bakılabilir.

> ssh -l anonymous sshSunucuIP

Böylelikle şifre falan girmeden hedef sisteme sızabiliriz. Tabi varsayılan ayarlar değiştirilmediği müddetçe...

9)

Farklı Portta Çalışan SSH Sunucularına Bağlanmak

Buraya kadar olan örneklerde SSH sunucusunun varsayılan olan port 22'de çalıştığını göz önünde bulundurarak bağlantı denemelerinde bulunduk. Eğer ssh servisi güvenlik nedeniyle başka porta kaydırıldıysa bu durumda bağlanmak için -p parametresini kullanmamız gerekir:

> ssh -l huzeyfe -p 200 www.enderunix.org

Output:

The authenticity of host 'enderunix.org (14.2.7.8)' can't be established.
RSA key fingerprint is 3f:98:e8:53:d7:62:1a:34:2e:57:39:47:f2:19:66:ea.
Are you sure you want to continue connecting (yes/no)?

10)

Uzak Sistemde Komut Çalıştırmak

> ssh -l huzeyfe www.enderunix.org ls /home

ya da

> ssh huzeyfe@enderunix.org ls /home

Output:

The authenticity of host 'cc.kou.edu.tr (1.2.7.8)' can't be established.DSA key fingerprint is a6:d6:35:52:75:66:63:15:5d:f6:76:b4:52:56:b4:64.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'cc.kou.edu.tr,1.2.7.8' (DSA) to the list of known hosts.
Password: XXX
6.0-BETA4-1-i386-disc2.iso
6.0-BETA4-i386-disc2.iso
6.0-BETA4-i386-disc2.iso.1
6.0-BETA4-i386-disc2.iso.2
Maildir
Manning,.Swt.JFace.in.Action.(2004).LiB(1).pdf
Manning.JUnit.Recipes.2005.pdf
Ev.
[…]

11)

Dosya Transfer Etmek

Günümüzde kullanılan en popüler dosya transfer aracı FTP’dir. SSH kullanarak hem ftp kolaylığında dosya transferi yapılabilir; hem de transfer edilen dosya şifrelenerek meraklı gözlerden korunmuş olur. SSH ile dosya transferinde temel olarak iki seçenek vardır; biri SCP(secure Copy) diğeri de sftp(Secure FTP). Temel bazı farklılıklar dışında her iki yöntem ile de yapılabilecekler aynıdır.

SCP Kullanım Örneği

Örneğin computerA makinesindeki deneme.txt dosyasını computerB makinesinin /usr/tmp dizinine kopyalamak için aşağıdaki kodlama kullanılabilir:

```
huzeyfe@computerA$ scp  deneme.txt  huzeyfe@computerB:/usr/tmp/
```

SFTP Kullanım Örneği

```
> sftp huzeyfe@enderunix.org

Connecting to enderunix.org...
Password:

sftp > get home

Fetching /home to /home/huzeyfe

sftp > help

[…]
```

NOT: Sftp ile sadece binary modda veri transferi yapılabilmektedir.

(Page 5-6)

12)

SSH Server'ı Farklı Porta Kaydırmak

Daha önce SSH Client'tan nasıl farklı porttaki SSH Server'a bağlanabileceğimizi görmüştük. Şimdi ise SSH Server'ın nasıl farklı porta kaydırıldığını öğreneceğiz. Bu yapılandırma ayarı için SSH Server'ın yapılandırma dosyalarına bir göz atalım.

```
> cd /etc/ssh
> ls -l
```

Output:

```
-rw-r--r--      1 root  root     1167 Eyl 17 2003 ssh_config
-rw------       1 root  root     2474 Eyl 17 2003 sshd_config
[…]
```

Sunucu için ayarlama yapacağımızdan dolayı inceleyeceğimiz dosya sshd_config olacaktır. Bu dosyayı açıp “#Port 22” satırını bulduktan sonra önündeki # karakterini kaldırırak dilediğimiz port numarasını 22 yerine girebiliriz. Ardından

```
> /etc/init.d/sshd restart
```

ile SSH Server'ı tekrar başlatarak yaptığımız değişikliği aktif hale getirmiş oluruz. sshd_config dosyasındaki diğer tüm seçenekleri ve yapılandırma ayarlarını görebilmek ve anlamlarına vakıf olabilmek için man sayfasına bakılabilir:

```
> man sshd_config
```

(Page 6-7)

13)

SSH Sunucusunda Root Olarak Oturum Açma

SSH sunucusuna root girişi yapabilmek için SSH sunucusunun yapılandırma dosyası sshd_config' deki “#PermitRootLogin” satırını “PermitRootLogin yes” olarak değiştirmek gerekmektedir. Böylelikle ssh sunucusuna root girişinin önü açılmış olur. Eğer kapatmak istersek “PermitRootLogin no” dememiz yeterlidir.

(page 7)

14)

SSH Sunucusuna Sadece Belirli Kullanıcıların Bağlanabilmesi

SSH sunucusuna sadece belirlediğimiz kullanıcı adlarına sahip kullanıcıların bağlanabilmelerini istiyorsak yine aynı dosya olan sshd_config'deki AllowUsers satırına kullanıcı adlarını eklememiz gerekir.

```
AllowUsers huzeyfe, ismail, murat
```

Kullanıcı adları içerisinde regExp (e.g. * , ? gibi) karakterler de kullanabilmekteyiz. Örneğin;

```
AllowUsers *ray
```

diyerek son üç harfi ray ile biten kullanıcı adlarına sahip client'ların ssh sunucusuna bağlanabilmelerine izin vermiş oluruz.

(Page 7)

15)

SSH Sunucusuna Belirli Kullanıcıların Bağlanmasını Engelleme

Belirli kullanıcıların ya da grupların ssh sunucusuna bağlanmalarını engellemek için yine sshd_config dosyasına bir göz atmamız gerekir. Bu dosyanın DenyUsers satırını bulduğunuz takdirde o satıra kullanıcı adlarını ekleyerek belirttiğiniz kullanıcı adlarının ssh sunucunuza girmesini engellemiş olursunuz. Örneğin;

```
DenyUsers fatih ali          // * ve ? karakterleri kullanıcı adı içerisinde kullanılabilmektedir.
```

Belirli bir grubu engellemek için ise sshd_config dosyasındaki DenyGroups satırına gelip engellemek istediğiniz kullanıcı gruplarının adını yazabilirsiniz. Örneğin;

```
DenyGroups root admin    // * ve ? karakterleri kullanıcı adı içerisinde kullanılabilmektedir.
```

(Page 7)

16)

sshd_config Dosyası Syntax Kontrolü

Üzerinde değişiklik yaptığınız sshd_config dosyasında yazım kurallarının dışına çıkmadığınızı aşağıdaki komut ile teyit edebilirsiniz:

```
> sshd -t
```

(page 7)

17)

SSH Oturumu Açma Sonrası Karşılama Mesajı

SSH Sunucusuna bağlanan kullanıcıyı karşılayan bilgilendirme ya da uyarı amaçlı mesaj değiştirilebilmektedir. OpenSSH için bu mesajın yer aldığı dosya şu dizindedir:

```
/usr/local/etc/warning.txt
```

(Page 8)

18)

Anahtar ile Oturum Açma

SSH'da kullanıcı adı ve şifre yerine anahtar aracılığıyla da oturum açılabilmektedir. Anahtar ile kimlik doğrulama adımları şu şekildedir:

a. SSH Client server'a falan filan kullanıcı adıyla bağlanmak istediğini belirtir.
b. Server client'tan gelen talebi alır ve client'ın kendini kanıtlaması için challenge mesajı gönderir.
c. Client ise private key'ini kullanarak gelen challenge verisini şifreler ve server'a cevap olarak gönderir.
d. Server kendinde olan private key'i ile orijinal challenge'i şifreler ve client'ın gönderdiği şifreli challenge ile aynı mı diye kontrol eder. Eğer aynılarsa server client'a giriş izni verir, diğer türlü giriş talebini reddeder.

NOT: Bu iletişim sürecinde hat üzerinden ne public ne de private key iletilmemiştir. Böylece meraklı gözler anahtarları sniff'lemeyeceğinden kaçak bir ssh girişi yapamayacaklardır.

19)

SSH Client Tarafında Anahtarları Oluşturmak

SSH anahtar çiftini oluşturmak için OpenSSH ile birlikte gelen ssh-keygen programı kullanılabilir. Bu programın -t parametresi ile RSA ya da DSA tipinde bir anahtar çifti oluşturulabilir:

> ssh-keygen -t rsa

Output:

Generating public/private rsa key pair.
Enter file in which to save the key (home/hefese/.ssh/id_rsa); *// Enter'la GEÇ.*
Enter passphares (empty for no passphrase):
Enter same passphares again:
Your identification has been saved in /home/hefese/.ssh/id_rsa.
Your public key has been saved in /home/hefese/.ssh/id_rsa.pub.
The key fingerprint is:
58:af:43:fd:b9:ba:26:d3:38:21:45:5d:dd:ac:d4:de hefese@home-fw.my.domain

İlk kalın yazının olduğu yerde direk ENTER yapılırsa ~ dizini altında anahtar çifti oluşturulacaktır. İkinci ve üçün kalın yazıların olduğu yerde ise belirlenen bir şifreyi girebilirsiniz. Böylece ~ dizini altında .ssh klasörü içerisinde id_rsa ve id_rsa.pub şeklinde biri private diğeri public olmak üzere iki anahtar dosyası oluşturulur.

Açık anahtarı SSH Server'a aktarmak için şifreli bir hat seçiminde bulunulmalıdır. Bunun için scp kullanılabilir. Bunun için ilgili SSH Sunucusunun home dizini altındaki .ssh klasörü içerisinde

authorized_keys dosyası oluşturulmalıdır.

~/.ssh/authorized_keys dos

Bu dosyanın içerisinde ssh-keygen ile client'ta oluşturulan id_rsa.pub içeriği aktarılmalıdır. Böylece artık SSH sunucusuna client anahtarlarıyla bağlanabiliriz.

(Page 8-9)

20)

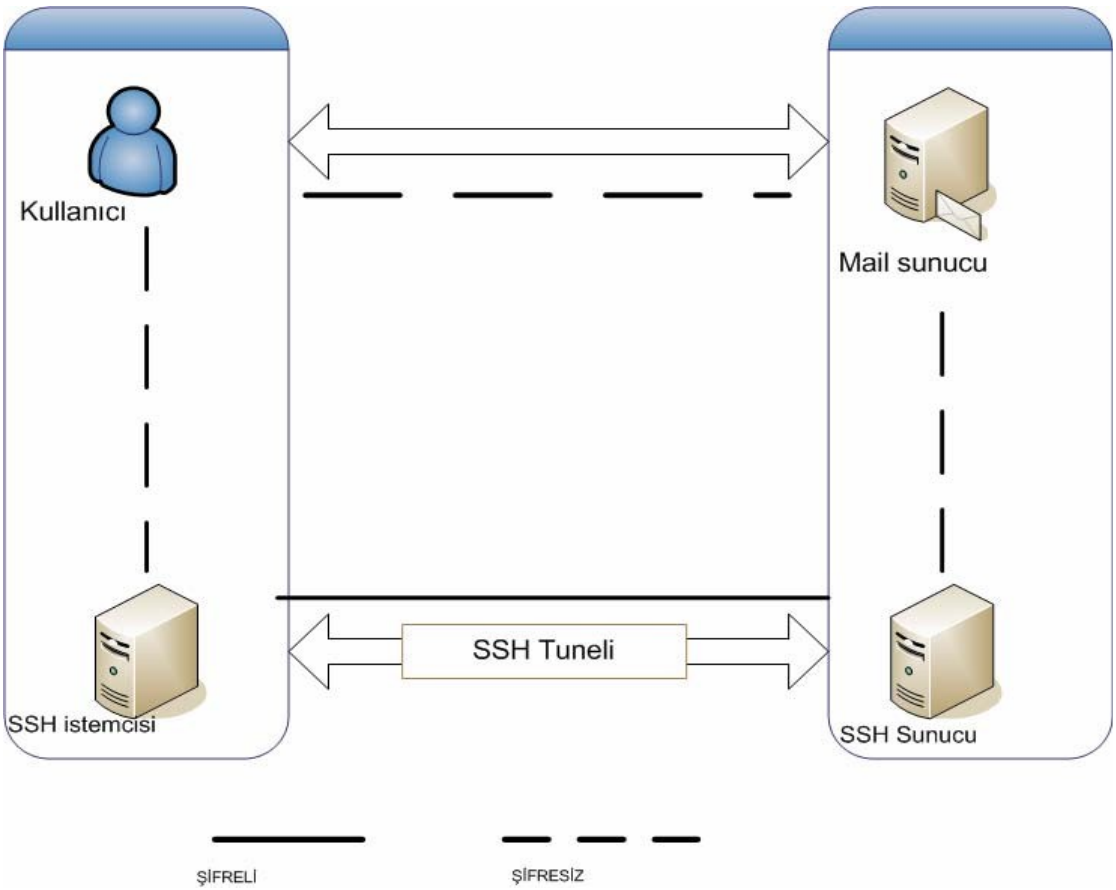
SSH Port Forwarding

Genellikle yapılan şey iletişimde şifreleme altyapısına sahip olmayan protokollerin (pop/imap/smtp vs...) şifreleme kullanan bir protokol aracılığı ile güvenli bir şekilde kullanılması yönünde bir tercihin seçimidir. Yani bir protokolü bir başka protokol aracılığıyla kullanmaktır. Buna forwarding, yani tünelleme denebilir.

Local Forwarding
Örneğin POP3 servisi TCP/110 üzerinden çalışır. İstemci ile POP sunucusu arasındaki iletişim clear text (şifrelenmemiş) olarak gerçekleşir. Uzaktaki POP sunucu ile istemci arasındaki trafiği SSH Port Forwarding kullanarak şifrelemek mümkündür. Şöyle ki öncelikle POP3 servisini kullanan yerel makinemizde 1024 – 65535 arası bir port seçelim. Mesela 5000 olsun. POP istemcisimizde kullandığımız pop sunucuyu uzak server'ın adresi yerine localhost yapalım ve portu da 5000 olarak ayarlayalım. Ardından SSH Client'a yönlendirme ayarını aşağıdaki gibi yapalım:

> ssh –L5000:localhost:110 mailSunucu(POP)

Bu komut sayesinde mail sunucusunda geçerli bir hesap ile ssh oturumu açılmış olur. Bu oturumda client localhost'un 5000. portu ile çıkış yapıp mail sunucusunun 110. portuna SSH ile bağlanmış olur. Böylece POP3 servisi ile POP3 sunucusu iletişimi ssh kanalıyla kurulmuş olur.



Adım adım inceleyecek olursak;

1. POP istemcisi yerel ağdaki 5000nci porta bağlanır.
2. Yerel ağdaki SSH istemcisi 5000nci porta gelen veriyi şifreler ve SSH aracılığıyla mail sunucusunun makinesine gönderir.
3. Mail sunucusu kendindeki ssh server aracılığı ile gelen veriyi çözerek 110ncu portuna iletir.
4. Bunun üzerine Mail sunucusu bir POP3 Response paketi oluşturur ve aynı güzergahı ters yönden takip ederek paketi gönderir. Böylece paket istemciye şifreli olarak ulaşır. SSH Client şifreyi çözer ve clear text olarak istemci cevabı (mail'ini) okur.

Remote Forwarding

Local Forwarding'ten farklı tünellemenin bu sefer client'tan değil server'dan yapılıyor olmasıdır. Yani Remote Forwarding'de server uzaktan client'a tünelleme yaptırtmaktadır. Şöyle ki

```
> ssh -R50000:localhost:110 istemciMakine
```

yukarıdaki komut istemciye ait makinenin 5000nci portu ile sunucu makinesinin arasında bir tünel oluşturur.

Benim NOT: Sanırım istemcideki POP3 servisinin server ve port tanımlarını yine istemcinin değiştirmesi bekleniyor. Diğer türlü sadece yukarıdaki kodla tünelleme yapılamaz. İlla ki POP3'ün server tanımı localhost, port nosu da 5000 şeklinde ayarlanması gerekir.

NOT: SSH ile sadece TCP tabanlı protokoller tünellenebilmektedir.UDP tabanlı protokoller tünellenememektedir. Ayrıca IP tabanlı olmayan protokoller de tünellenememektedir. Bu durum SSH'ın VPN bağlantıları karşısındaki bir dezavantıdır.

(Page 10-12)

21)

SSH Server Güvenliği Hakkında

Son zamanlarda SSH protokolüne karşı yapılan saldırılar artmıştır. Eğer bir SSH sunucu çalıştırıyorsanız sistem loglarında sunucunuzdaki SSH servisine yapılan atakları görebilirsiniz. Bu ataklar genellikle SSH sunucudaki zayıf parolalarla korunmuş sistem hesaplarını ele geçirmek için yapılır. Bu tip ataklara birçok farklı şekilde önlem alınabilir. Kompleks çözümlere kaçmadan yapılacak birkaç basit ayarlar bu tip saldırıların %90'na karşı doğal koruma sağlanmış olur. Bu doğal korumalar şunlardır:

a. Kullandığınız OpenSSH sürümünün güncel olmasına özen gösterin.
b. Port 22 olmazsa olmazınız değilse SSH Server portunu 22'den farklı bir porta alın. Mesela doğum tarihiniz.
c. Sisteme erişim yetkisi vermek istediğiniz kullanıcıları yapılandırma dosyasında belirtin.
d. Sisteme root olarak erişim izni vermeyin.
e. Mümkünse sisteme parola ile girişi yasaklayıp erişimleri anahtarlar aracılığıyla yapmaya çalışın.
f. SSH erişimini tüm internete açmayın. Sadece belirli IP'lere erişim açın. Bu işlemi herhangi bir firewall kullanarak ya da hosts.allow/hosts.deny dosyalarını kullanarak yapabilirsiniz.

(Page 13)