Contents

Figure 1: ULAKBIM

Sertifika Otoritesi (CA) Kurulumu ve Anahtar Yönetimi

[TOC]

Bu dokümanda, Ahtapot bütünleşik güvenlik yönetim sisteminde kullanılan SSH anahtarlarını yöneten Sertifika Otoritesi (CA - Certificate Authority) sisteminin kurulması ve anahtar imzalama prosedürü anlatılmaktadır.

Gereken : Pardus Temel ISO' dan kurulumu tamamlanmış bir sunucu.

Önemli Uyarılar

- Kurulacak sunucu, PKI (Public Key Infrastructure) yapısının omurgasını teşkil edeceğinden yüksek düzeyli korunacak sistemler arasında yer almalıdır.

Sertifika Otoritesi Temel Anahtarı Oluşturma

UYARI : Aşağıdaki adımların çalıştırılacağı sistem "**Ahtapot Sertifika Otoritesi Sunucusu**" olmalıdır.

- Pardus Temel ISO' dan Pardus kurulumu tamamlandıktan sonra sistemde tanımlı bir kullanıcı ile (tercihen root) Ahtapot Sertifika Otoritesi olacak sunucu sistemine bağlantı sağlanır. ssh-keygen kullanılarak standart bir SSH anahtarı oluşturulur. Bu anahtar dosya ismi olarak "**ahpapot_ca**" ön eki ile oluşturulur. SSH anahtarı oluşturulurken kullanılan şifrenin özenle saklanması gerekmektedir.

```
ahtapotops@ahtapot:~ ssh-keygen -f ahtapot_ca
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ahtapot_ca.
```

```
Your public key has been saved in ahtapot_ca.pub.
The key fingerprint is:
ad:78:1c:b3:32:09:0d:d0:4e:3a:37:8a:fd:81:73:fd [MD5] ahtapotops@ahtapot.com
The key's randomart image is:
+--[ RSA 2048]----+
|  ..             |
|   .o            |
|   +.            |
|  o +o   .       |
| o =.o. S .      |
|. = o..+ =       |
|   + .=.=        |
|    .  +E        |
|                 |
+--[MD5]+
```

- Yukarıdaki adım ile 2 adet dosya oluşturulur. Bu dosyalardan "**ahtapot_ca**" özenle korunması gereken gizli anahtar (Private Key), "**ahtapot_ca.pub**" dosyası ise her yere dağıtılabilen açık anahtar (Public Key) olarak kaydedilir.

```
ahtapotops@ahtapot:~ ls -al
total 8
drwxr-xr-x 1 ahtapotops users   52 Oct 29 10:44 .
drwxr-xr-x 1 ahtapotops users  274 Oct 29 10:34 ..
-rw-r--r-- 1 ahtapotops users 1766 Oct 29 10:44 ahtapot_ca
-rw-r--r-- 1 ahtapotops users  405 Oct 29 10:44 ahtapot_ca.pub
```

İmzalanacak Kullanıcı Anahtarı Oluşturma

- Ahtapot sistemini yönetecek olan ahtapotops kullanıcısı için başka bir anahtar oluşturulur. Bu anahtar otomatik sistemlerle kullanılacağından ötürü şifre verilmeden oluşturulmalıdır.

```
ahtapotops@ahtapot:~ ssh-keygen -f ahtapotops
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ahtapotops
Your public key has been saved in ahtapotops.pub
The key fingerprint is:
51:5c:40:e8:6e:3d:b4:6d:8f:49:7f:b3:ad:de:51:d6 [MD5] ahtapotops@ahtapot.com
The key's randomart image is:
+--[ RSA 2048]----+
|         ++o.    |
```

```
|         ...      |
|        ..        |
|         ...     .|
|        .So o    E|
|         o + +  ..|
|        .   + = . |
|             o oo+|
|              .o+=|
+--[MD5]+
```

- Yukarıdaki adım ile 2 adet dosya oluşturulur. Bu dosyalardan "**ahtapotops**" özenle korunması gereken gizli anahtar (Private Key), "**ahtapotops.pub**" dosyası ise her yere dağıtılabilen açık anahtar (Public Key) olarak kaydedilir.

NOT : Bu yöntem kullanılarak aşağıdaki kullanıcı listesi için farklı anahtarlar oluşturulmalıdır.

```
* git
* myshook
* gdyshook
* fw_kullanici
```

Sertifika Otoritesi Anahtarı ile Kullanıcı Anahtarı İmzalama

- Oluşturulan kullanıcı anahtarlarının **ahtapot_ca** ile imzalanması gerekmektedir.

```
ahtapotops@ahtapot:~ ssh-keygen -s ahtapot_ca -I ahtapotops@ahtapot.com -n ahtapotops -O source
Enter passphrase:
Signed user key kaptan-cert.pub: id "ahtapotops@ahtapot.com" serial 0 for ahtapot valid forever
```

- Yukarıdaki adım ile "**ahtapotops.pub**" dosyası sadece 10.0.7.0/24 network bloğundan, sadece ahtapotops kullanıcısı olarak bağlanacak şekilde kısıtlandırılarak imzalanır. Ayrıca SSH yaparken port forward etme, X11 protokolüyle erişim gibi özellikler de kısıtlanır. Bu dokümanın sonunda SSH anahtarı imzalanırken kullanılabilecek olan tüm opsiyonlar detaylandırılmıştır.
- Yukarıdaki komutta "**-V YYYYMMDDHHMMSS**" opsiyonu kullanılarak belli bir süreye kadar geçerli olma ayarlaması da yapılabilir. Örneğin, yukarıdaki anahtarı 1 Ocak 2016 'dan 1 Ocak 2018'e kadar geçerli olacak şekilde oluşturmak için:
- İsteğe bağlı olarak **-O source-address=10.0.7.0./24** parametresi komuttan çıkarılabilinir.
- Oluşan imzalanmış anahtar dosyasının adı "**ahtapotops_cert.pub**" dir. Bu dosyada ki imzalanmış kısıtlamalara göz atmak için aşağıdaki komut

kullanılır.

```
ahtapotops@ahtapot:~ ssh-keygen -Lf ahtapotops-cert.pub
kaptan-cert.pub:
        Type: ssh-rsa-cert-v01@openssh.com user certificate
     Public key: RSA-CERT 92:4b:a6:c5:8c:60:c5:c1:ab:37:a9:6f:64:22:1b:9e
     Signing CA: RSA 68:1c:65:34:85:10:d7:56:db:99:c5:31:43:4d:e5:24
        Key ID: "ahtapotops@ahtapot.com"
        Serial: 0
        Valid: from 2016-01-01T00:00:00 to 2018-01-01T00:00:00
        Principals:
                ahtapotops
        Critical Options:
                source-address 10.0.7.0/24
        Extensions:
                permit-pty
                permit-user-rc
```

NOT : Bu yöntem kullanılarak git kullanıcısı için de imzalama işlemi gerçekleştirilmelidir.

Sertifika Otoritesi Anahtarı ile Kısıtlı Anahtar İmzalama

- fw_kullanici anahtarının sadece FirewallBuilder uygulamasını çalıştırması için **-O force-command** parametresi ile çalıştıracağı komut yazılmalıdır.

```
ahtapotops@ahtapot:~ ssh-keygen -s ahtapot_ca -I ahtapotops@ahtapot.com -n ahtapotops -O permit
Signed user key fw_kullanici.pub: id "ahtapotops@ahtapot.com" serial 0 for ahtapotops valid for
```

- Oluşturulan anahtar bağlantı kurduğu sunucuda "**ahtapotops**" kullanıcı yetkilerine sahip olacak, X11 kullanabilecek ve **sadece** FirewallBuilder uygulamasına erişebilecektir. FireallBuilder uygulamasını başlatma komutu olan "**/var/opt/gdysgui/gdys-gui.py**" dışında herhangi başka bir komut **çalıştıramaması** SSH ve PKI alt yapısı tarafından garanti altına alınır.
- Oluşturulan anahtarın kısıtlamalarına göz atmak için :

```
ahtapotops@ahtapot:~ ssh-keygen -Lf fw_kullanici-cert.pub
fw_kullanici-cert.pub:
        Type: ssh-rsa-cert-v01@openssh.com user certificate
     Public key: RSA-CERT 34:cc:67:20:2f:01:74:2f:e7:3f:f8:ca:a9:3a:ec:77
     Signing CA: RSA 62:56:a2:64:62:67:97:94:d2:9e:7b:5d:d9:35:95:0c
        Key ID: "ahtapotops@ahtapot.com"
        Serial: 0
        Valid: forever
        Principals:
                ahtapotops
```

```
Critical Options:
        force-command /var/opt/gdysgui/gdys-gui.py
Extensions:
        permit-X11-forwarding
        permit-agent-forwarding
        permit-port-forwarding
        permit-pty
        permit-user-rc
```

- Yukarıdaki yöntem kullanılarak gdyshook ve myshook anahtarları da imzalanmalıdır.

gdyshook için:

```
ahtapotops@ahtapot:~ ssh-keygen -s ahtapot_ca -I ahtapotops@ahtapot.com -n ahtapotops -O no-por
Signed user key gdyshook.pub: id "ahtapotops@ahtapot.com" serial 0 for ahtapotops valid forever
```

myshook için:

```
ahtapotops@ahtapot:~ ssh-keygen -s ahtapot_ca -I ahtapotops@ahtapot.com -n ahtapotops -O no-por
Signed user key myshook.pub: id "ahtapotops@ahtapot.com" serial 0 for ahtapotops valid forever
```

komutları çalıştırılmalıdır.

Sertifika Otoritesi Anahtarı ile Kullanıcı Erişim Kısıtlama Ayarları

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
force-command=command	Forces the execution of command instead of any shell or command specified by the user when the certificate is used for authen-

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
no-agent-forwarding	Disable ssh-agent(1) forwarding (permitted by default).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
no-port-forwarding	Disable port forwarding (permitted by default).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
nopty	Disable PTY allocation (permitted by default).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
no-user-rc	Disable execution of ~/.ssh/rc by sshd(8) (permitted by default).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
no-x11-forwarding	Disable X11 forwarding (permitted by default).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
permit-agent-forwarding	Allows ssh-agent(1) forwarding.
permit-port-forwarding	Allows port forwarding.

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
permit-pty	Allows PTY allocation.
permit-user-rc	Allows execution of ~/.ssh/rc by sshd(8).

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
permit-x11-forwarding	Allows X11 forwarding.

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.
source-address=address_list	Restrict the source addresses from which the certificate is considered valid. The address_list is a comma-separated list of one or more address/netmask

clear	Clear all enabled permissions. This is useful for clearing the default set of permissions so permissions may be added individually.

SSL Anahtar Oluşturma

- Rsyslog veya nxlog kullanılarak client-server arasında log gönderimi için öncelikle "**CA sunucusu**" üzerinde "**openssl**" kullanılarak "**rootCA**" oluşturulmalıdır.

```
openssl genrsa -out rootCA.key 2048
openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.pem
```

- Log gönderecek veya log'u alacak makineler için openssl kullanılarak anahtarlar oluşturulur. İkinci komut çalıştırıldığında gelen soru ekranlarında "**Common Name**" satırına ilgili makinanın **FQDN** bilgisi girilmesi zaruridir.

```
openssl genrsa -out client_fqdn.key 2048
```

```
openssl req -new -key client_fqdn.key -out client_fqdn.csr
```

- Oluşturulan anahtarlar rootCA ile imzalanır.

```
openssl x509 -req -in client_fqdn.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out cli
```

NOT: İstenilen durumlarda her makine için tek veya ayrı anahtarlar oluşturularak Sertifika Otoritesi sunucunda imzalanılabilir. Her makinede aynı anahtar kullanılacak ise, makinelere taşınırken ilgili "**crt**" ve "**key**" dosyasının adı ilgili makinaının FQDN' i ile değiştirilmelidir.

Sayfanın PDF versiyonuna erişmek için buraya tıklayınız.