KERESTECİLER TEŞKİLATI

TOMRUK GÖREV GRUBU

E.N.İ.K. OPERASYONLARI BİRİMİ

ELEKTRONİK İSTİHBARAT ve *KEŞİF*

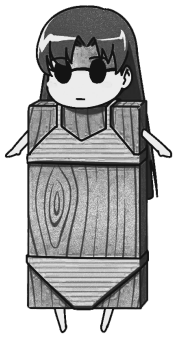
KONU:

GPG Kullanımı ve Açık Anahtar Kriptografisi

İÇERİK:

1. *Diffie-Helman Anahtar Anlaşması*

(Not: “x = m * n + y” ise, “x mod n = y” ve “x mod m = y” dir. Kısaca, “m odülüs” kalanı bulmak demektir.)



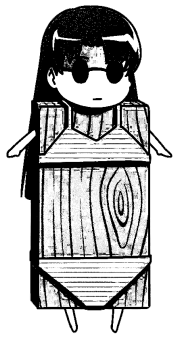
Bob

P=23
G=5

AÇIK KANAL
(Herkes Görebilir)

1. Bob ve Alice ortak parametreler belirler
P=23 ve **G=5**

P=23
G=5



Alice

b = 3

2. *Bob* şahsi anahtarı (**b**) ile ortak parametreleri (**P** ve **G**) birleşirir ve oluşturduğu açık anahtarını (**B**) *Alice*'e gönderir

$B = G^b \bmod P = 10$
$B = 5^3 \bmod 23 = 10$

a = 4

3. *Alice* şahsi anahtarı (**a**) ile ortak parametreleri (**P** ve **G**) birleşirir ve oluşturduğu açık anahtarını (**A**) *Bob*'a gönderir

$A = G^a \bmod P = 4$
$A = 5^4 \bmod 23 = 4$

4. *Bob*, *Alice*'in açık anahtarı (**A**) ile kendi şahsi anahtarını (**b**) birleştirir

$S = A^b \bmod P = 18$
$S = 4^3 \bmod 23 = 18$

5. *Alice*, *Bob*'un açık anahtarı (**B**) ile şahsi anahtarını (**a**) birleştirir

$S = B^a \bmod P = 18$
$S = 10^4 \bmod 23 = 18$

Nihayetinde kendileri arasında gizli bilgi (**S**) aktarımını başkalarının haberi olmaksızın gerçekleştirmeyi başardılar

Bu tür haberleşmenin gerçekleşeceğinden bir başkasının haberi olsa bile tersine döndürmesi çok zordur. Gerçek uygulamalarında bazı parametrelerinin çok büyük asal sayılar olabileceği ve “Diffie-Helman Anahtar Anlaşmasının” geliştirilen yöntemlerden sadece biri olduğu bilinmelidir.

2. *Anahtar Yönetimi*

----2-a. *ASCII-Armored Anahtar yapısını tanıyalım* ----------

Başlangıcını ve sonunu gösteren belirteçler bulundurur.

Belirteçleri olmadan eldeki veri anlamsızdır.

Bunun gibi ASCII formatındaki anahtarlar bir uçbirime yapıştırılabilir veyahut “.asc” / “.key” uzantılı bir dosyanın içine yerleştirilerek işlenebilir.

“.asc” spesifik olarak ASCII formatında olduğunu belirtmek için kullanılır.

“.key” ise genel bir kullanımdır, ASCII formatında da olabilir.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGcL9MsBEACqFYxnD3vkDp4oNjLMmzjtdacBrEi5tVE6nVmiil4qn37pkw/X
zoRo9PmZcKhnkAnBfzQi17RIELT7Txd+yaY41kUzcx0DLz0jc4CKOa/bCpcqOPRP
GTSlvEJCSNNuAK1LwORl+fwPcpw1Sayjnacew8mxwp/LkqBEO1Ryr0o993oNYOPs
MrFw9DON8tYvUlSp4EiUHdI6ixMHd+cd7QemM9/qCSTqz9h2mr2k5vBh4/rautSl
CBFCTFO/vmKu7eXjgKD5vchI7Fi9J+6FCDL7hE7j6C/VrCbUvzkfuDvOGwDNdHoo
zkZZ3GxW/I0U5JFYKJNNmw45d+rLbMdsdgrxiURqoqyJ5Tz6DTbxSdsVwBlaoXph
NA6PpIh0Hm9C3qA21XGIuZR1F4rhudKfx+nbo3UihrhifmtkPYqt+WXlRyt1ltrs
KwsHlgAESdFLeoigw266W/0OKREMijv/46OrBwU+8X8C8S/gTrFeEPIKV2VjmMWn
qW6uHX8VYMydUxHd4EkS3R07gYWvAYoPnI6qKWdPSF8RTZh/oBLlqZ7AQU4ymR75
3+v8PUV+zs9ZekAikAOD5AITPEO9ElfTz0wEj1G6ATNIRCee1e0AsfygnK+9B0sJ
bdXjIUj/kmt8mkgTfgQwpNKik4sfMzwsXe6Dt5ZEdNMrtjEKsL/qHnECbwARAQAB
tCRCaWxsIEdhdGVzIDxiaWxsZ2F0ZXNAbWljcm9zb2Z0LmNvbT6JAlQEEwEKAD4W
IQRQUS8Y6rCE0bCqo5aw+s9/9nDF4AUCZwv0ywIbAwUJDwmcAAULCQgHAgYVCgkI
CwIEFgIDAQIeAQIXgAAKCRCw+s9/9nDF4A/KD/9JYWn+gMidq/hCqmuLT8DnbTi1
tgIQQ9RR03Bk3pkXy9lSNE4Ove5zCBDDYDVCcm5VUhQLztNI66RgquPuaZVR/ZYk
WlCeBFJ86kwnD1HzwkfJ1/TIc77eFfdcwinnvYtbAq9q90hSWfj/qY6eJO6ZCcUO
gkdhu6LK2Do5AdsY9YV9yI65ej/Q9SwJEBQB0bE+UVbKi/DSklGSREEkjP5mhbkk
YRABWv9vA6aTMBUv9UPUm9IBjCkw9E7YtUMErO/F4MXX4iQtbHWLGGSpaJeWRDC3
lqNLMzI4iSU2cAbgTJuplI2Cx6iWQNihpD29151f9wqOfgDa/8OxIVvlqMjRxWCJ
mxbQctHvghdIJhV/7W95d25g47L/EFYzEdDc8bB81SimyQv7QTSyB8nJ/OdK9r9W
je25DOCyDvhbGYfvsmJq9Qk5BjQl8VAuhLQvhVixGpogK1jWVWZ0r9mW82qpZRSD
z+DQnm7Jd74EmWfZjStK+CM30tYFsaDbcuNvpLJg8v203y0e0uJzWvCpIZmQqJfK
ES/FZogIw4YAq4dMRTi/DcpsPhhvUEKiKmmzr/kqTfqICNt5vTBmO3IWO8d4y+c0
6j9ui/KR5tCgmmgSpDSDHKcCXKk3lp43d8BJLTo9ccxFcb6uIA1lcut0wOb2oWSB
rQQok6SwDaMA3fZykLkCDQRnC/TLARAAzp6F2CrhljXogGD0mFQmHe24vaO//dRJ
S1hRQE4VmhfZbfkT49+C5PlBUNywblF+GzkxapxsKfR0nh4V8frI9EkBE066reeR
ZFye3KoFyVoUWWWrIOiAwGiAaGi3srBYhwo0leMQTyuyOWNVVPW5iZ2r7hcaDjNa
+OiBhVcqrN3OPMOFRAF/yosMifGApxcKcisoOi7gPIZdQ/kXxAOGq9R9Y8AHf8ug
xY+NvpCy8rpw/g7B111yXVhH9QkUBFsjSXuBE5T0bz0H5qmVz0VAOQn3u3hN2SKv
KbVUBblF2iCMqhL3WpenW7Djxl7S6ihS1yAe1n6daWItMLsmRwyLv10coy2HaDih
JADjrVtMHKnmgGZD30AFrOXn1vhwPRLwzzBwnG441Z7nZVASE6T6FA/DBes0GWCo
zaDdmysJgYkLGubLwGfkEuR7VllbM4oOoRcGDjDU3XceluCYtz9La3ReGHCsH2j3
eVIo/5Mur1t9fi6JD/mzdwAa9f4SOdlQyMqwhgjCvvQUp5uXKgwJo2cz+36jFPzd
N0btgiZLJw/E8y04Fd2iP04/lMSHQZxUybNfrWe8m6ovMJdbJTcXrzenBWR2pX21
OwgsH2Zm8yS6fsPBVFkArU0XbpRVv8MAPY3p/tq6EcWdRqYCOI4waJ9Z1c5J7YSI
uwbgL8mWeKEAEQEAAYkCPAQYAQoAJhYhBFBRLxjqsITRsKqjlrD6z3/2cMXgBQJn
C/TLAhsMBQkPCZwAAAoJELD6z3/2cMXgK3oP/jQ8fD+4m+4/gemPGswoAmQ9SLnX
psNEKeDbFHvC5tCoQud/j7TuakhZoz7bLonKNrVrPwXH0IN2cS0T8dcxNR/wiLCK
W5fCYCHa4+FkxPre0s6gviShZXYr8rQX8WR03mYgABWJ3eoMKD08U6QrtBiwGXQc
ck0oU/3rxnFaGSCY50WzMMt+DEVpekryS/8lMm6/L/QiHFXvfPaFeRRCs896XyZG
uGuo1veF5Jf5uSFGZyjpXQAVxfHCc+4VFA5iCyDT94QWdwbjnmIK2WOX6OSXGpN3
XSqTLVva9KHDNIZ2vqc+jpvo49KB6SC8AL+cE2ZHqasj4q29MxSOFpVpLDMuxL+l
nWmA6M1vRLc2x7CVPqCrN7mSt0Nh6fOvrd4GH/SmKm5jV5eVccE2meSKqUmeC5Bh
0fvOnBQyKOALirjcObMK8ex/3Fa/J9THNpGAZU/ZrLEaaAL5Xym2uYjB7qE3+FdL
yjoIHg78gan6ZNbfOhevwH4XxgmrklFGuywDGcQZFP5+PY7jegVz3fW/4FUDV7Ut
ssbaQNEW7tGaCzHiXCMoTEZq+meI4jjqXUUv9iLHHQEjCgfSRwyAWEO7torjag8u
vjT0HCcHlKlPmQu9SZrCHZq5sJH9raDLtECNdJJeTFpbTiEp1lWAS/MKKHnsVaZL
mQLNheMVv+5urZmb
=Gjks
-----END PGP PUBLIC KEY BLOCK-----
```

Bu kılavuzda örnek almak üzere oluşturduğum anahtarımın açık anahtarının ASCII biçiminde görüntülenebilir hali.

Metin olarak rahatlıkla kopyalanıp yapıştırılabilir ve görüntülenebilir.

Bu yüzden paylaşılırken sıklıkla bu şekilde biçimlendirilir.

----2-b. *Bir açık anahtarı kişisel anahtarlığa ekleme* ----------

2-b-1. *Dosya formatında işleme*

(Halihazırda bir anahtar dosyanız var ise **ON1** işaretinden devam edin)

Favori metin editöründe yeni bir dosya açın ve örnek olarak 3. sayfada gösterilen ASCII PGP açık anahtarını içine yapıştırın.

Not: *Başlangıç ve bitiş belirleçlerini "[-----BEGIN PGP PUBLIC KEY BLOCK-----] ve [-----END PGP PUBLIC KEY BLOCK-----]" dahil ettiğinizden emin olun.*

Dikkat edilmesi gereken bir diğer şey ise "[-----BEGIN PGP PUBLIC KEY BLOCK-----]" devamında boş bir satır bulundurmalıdır. Boş bir satır yoksa yine de işlem başarıyla gerçekleşir ancak formatın düzgün olmadığına dair bir uyarı alırsınız.
Bu kılavuzda boş satırın bulunmama nedeni PDF dosya formatında boş satır kopyalanamamasıdır.

Oluşturduğunuz dosyayayı kaydederken dilediğiniz ismi verin ve uzantısını ".asc" veya ".key" olarak ayarlayın.

ON1 ----------

Bir Uçbirim/Komut İstemi penceresi açın ve konumunuzu anahtarı olduğu dizinle değiştirin.
Dizin değiştirdiğinizde alttaki komutu paremetreleriyle kaydettiğiniz anahtarın TAM dosya ismi ile değiştirerek girin ve çalıştırın:

```
gpg --import anahtar.asc
```

Not: *Anahtarın içeriğini incelemek için şu komutu çalıştırabilirsiniz*

```
gpg anahtar.asc
```

2-b-2. *Uçbirim/Komut İstemi girdisinden işleme*

Örnek olarak **3. sayfadaki** ASCII PGP açık anahtarını kopyalayın ve bir Uçbirim penceresi açın.

Not: ***Başlangıç ve bitiş belirleçlerini*** *“[-----BEGIN PGP PUBLIC KEY BLOCK-----]* ***ve*** *[-----END PGP PUBLIC KEY BLOCK-----]”* ***dahil ettiğinizden emin olun.***

Dikkat edilmesi gereken bir diğer şey ise *“[-----BEGIN PGP PUBLIC KEY BLOCK-----]”* ***devamında boş bir satır bulundurmalıdır.***

Bu kılavuzda boş satırın bulunmama nedeni PDF dosya formatında boş satır kopyalanamamasıdır.

Şu komutu parametreleri ile beraber girin ve çalıştırın:

```
gpg --import
```

Dikkat: *Sonraki adımlar işletim sistemine göre değişkenlik gösteriyor*

Kopyaladığınız anahtarı yapıştırın ve Uçbirim penceresi odakta iken Enter’a ardından ‘CTRL+D tuş kombinasyonuna basın (UNIX)/Komut İstemi odakta iken CTRL+Z tuş kombinasyonuna ardındanda Enter tuşuna basın (Windows)’.

EK 1: Düzenlemekle uğraşmak istemiyorsanız önce
“[-----BEGIN PGP PUBLIC KEY BLOCK-----]”
kısmını kopyalayıp yapıştırdıktan sonra ENTER tuşuna basın.

Sonrasında kalan metni kopyalayıp-yapıştırın ve önce ENTER tuşuna sonra ‘CTRL+D tuş kombinasyonuna basın (UNIX)/ CTRL+Z tuş kombinasyonuna sonra Enter tuşuna basın (Windows).

2-b-3. *Bir Keyserverdan ekleme*

2-b-3-a. *Keyserverlar*

Anahtarını eklemek istediğiniz kişi ASCII-Armored olarak paylaşmak yerine sadece PGP anahtarının parmak izini paylaştıysa bir keyserverdan arayıp ekleyebilirsiniz.

PGP parmak izi örneği: **50512F18EAB084D1B0AAA396B0FACF7FF670C5E0**
veya: **5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0**
PGP: 5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0 şeklinde görebilirsiniz. Bizim için önemli kısım **PGP:**'den sonra gelen 4'lü 10 grup.

Bazı Keyserverlar	
keys.openpgp.org	E-Mail doğrulaması gerektirir
pgp.mit.edu	Standart
pgp.surf.nl	Standart
keys.mailvelope.com	Standart
keyserver.ubuntu.com	Standart
keyring.debian.org	Debian geliştirici ekibinin anahtarları

GPG varsayılan olarak keys.openpgp.org adresini kullanır.

Debian ve OpenPGP keyserverları dışındakilerde tanımadığınız herhangi bir E-Mail'e kayıtlı olan anahtarın doğruluğundan emin olamazsınız. Parmak izi önemlidir.

OpenPGP, E-Mail üzerine kayıtlı bir anahtar eklemeye çalıştığınızda bu E-Mail adresine doğrulama kodu gönderecektir. Bu yüzden o anahtarın gerçekten E-Mail adresiyle ilişkili olduğunu bilebilirsiniz.
Debian ise sadece kendi geliştiricilerinin anahtarlarına izin verdiği için güvenilirdir.

2-b-3-b. *Keyserver Kullanımı*

2-b-3-b-1. *Anahtar Arama*

Öncelikle kullanacağınız keyserveri seçin, ne seçtiğinizi bilemeyeceğimden buna 'SERVER' diyelim. Elinizde bulunan E-Mail 'MAIL' ve İsim 'UID' olsun.

Arama yapmak için şu komut ve parametre grubunda 'SERVER', 'MAIL' ve 'UID'yi kendi seçtiklerinizle değiştirin:

```
gpg --keyserver SERVER --search-keys MAIL
```

veya

```
gpg --keyserver SERVER --search-keys UID
```

Numaralandırılmış sonuçlardan istediğinizi seçin ve açık anahtar kişisel anahtarlığınıza eklenecektir.

EK 1: ***Dilerseniz parmak izini de aratabilirsiniz. Parmak izine 'PRINT' diyelim.***

Arama yapmak için şu komut ve parametre grubunda 'PRINT'i elinizdekiyle değiştirin:

(5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0'ı parmak izi örneği olarak kullanabilirsiniz.)

```
gpg --keyserver SERVER --search-keys PRINT
```

Anahtarı eklemek için eşleşen sonucu seçin

Not: ***keys.openpgp.org'un katı standartları nedeniyle oradan doğrulanmamış anahtarları ekleyemezsiniz.***

2-b-3-b-2. *Anahtar Alma*

Parmak izini bildiğiniz bir anahtarı aramak yerine doğrudan ekleyebilirsiniz.

Öncelikle kullanacağınız keyserveri seçin, ne seçtiğinizi bilemeyeceğimden buna ‘SERVER’ diyelim. Elinizde bulunan PGP anahtarının parmak izi de ‘PRINT’ olsun.

(**5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0**’ı parmak izi örneği olarak kullanabilirsiniz.)

Alıp getirmek için şu komut ve parametre grubunda ‘SERVER’ ve ‘PRINT’i kendi seçtiklerinizle değiştirin:

```
gpg --keyserver SERVER --receive-keys PRINT
```

Eşleşen anahtar kişisel anahtarlığınıza eklenecektir.

Not: *Anahtar parmak izi 4’lü gruplar halinde,*
5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0 *gibi bölüştürülmüşse bütün olduğunu belirtmek için PRINT’i kesme veya tırnak işaretleri içine alın*
'5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0'
veya
"5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0" *olarak.*

2-b-3-b-3. *Anahtarı Alıp Getirme*

Bir URL’de depolandığını bildiğiniz anahtarı şu komut ve parametreleri çalıştırarak ekleyebilirsiniz:

```
gpg --fetch-keys URL
```

---- 2-c. *Kişisel anahtarlığınızı kontrol edin* ----------

Bulundurduğunuz anahtarları listelemek için şu komut ve parametreleri çalıştırın:

```
gpg --list-keys
```

Çıktısında şuna benzer girdiler sıralanacaktır:

```
pub   rsa4096 2024-10-13 [SC] [expires: 2032-10-11]
      50512F18EAB084D1B0AAA396B0FACF7FF670C5E0
uid           [ unknown] Bill Gates <billgates@microsoft.com>
sub   rsa4096 2024-10-13 [E] [expires: 2032-10-11]
```

Burada “pub   rsa4096 2024-10-13 [SC] [expires: 2032-10-11]**” satırı açık anahtarın** (pub) **şifreleme algoritmasını** (rsa), **bit uzunluğunu** (4096), **oluşturulma tarihi** (2024-10-13), **imzalama ve sertifika oluşturma için kullanılabileceği** ([SC] **Signing and Certificate Generation) ve ne zaman kullanım dışı kalacağını** ([expires: 2032-10-11]) **gösteriyor.**

Hemen altındaki “50512F18EAB084D1B0AAA396B0FACF7FF670C5E0**” satırı anahtarın parmak izini gösteriyor. Ek olarak parmak izinin son 8 byte’ı (son 16 karakter) long KeyID olur “**B0FACF7FF670C5E0**” veya “**0xB0FACF7FF670C5E0**” olarak gösterilebilir. Bunu kontrol etmek için listeleme komutuna “**--keyid-format=long**” parametresini ekleyebilirsiniz.**

“uid [unknown] Bill Gates <billgates@microsoft.com>” **satırı geçerlilik durumu** ([unknown]), **isim** (Bill Gates) **ve e-mail’i** (<billgates@microsoft.com>) **gösteriyor.**

“sub rsa4096 2024-10-13 [E] [expires: 2032-10-11]” **satırı alt anahtarın** (sub) **şifreleme algoritmasını** (rsa), **bit uzunluğunu** (4096), **oluşturulma tarihini** (2024-10-13), **şifreleme için kullanılabileceğini** ([E] **Encryption) ve ne zaman kullanım dışı kalacağını** ([expires: 2032-10-11]) **gösteriyor. Alt anahtar parmak izini görüntülemek için “**--with-subkey-fingerprints**” parametresini ekleyebilirsiniz.**

2-c-1. *Bir anahtarı silme*

Bir anahtarı silebilmek için öncelikle onu tanımlayacak bilgiye ihtiyaç duyarsınız, bu anahtarın parmak izi veya KeyID'si olabilir. Parmak izi değerine 'PRINT', KeyID değerini aynen 'KEYID' olarak alalım.

Anahtarlığınızdan bir anahtarı silebilmek için şu komutu değiştirdiğiniz parametreleri ile çalıştırmanız gerekir:

```
gpg --delete-key PRINT
```

veya

```
gpg --delete-key 0xKEYID
```

veya

```
gpg --delete-key KEYID
```

Komutu çalıştırdığınızda karşınıza şöyle bir istem gelir:

```
pub  rsa4096/B0FACF7FF670C5E0 2024-10-13 Bill
Gates <billgates@microsoft.com>

Delete this key from the keyring? (y/N)
```

Silmek için Y **yazın ve Enter'a basın**

EK 1: ***Birden fazla anahtarı aynı anda silmek isterseniz "--delete-key" yerine "--delete-keys" parametresini kullanın ve birden fazla anahtar parmak izi veya KeyID yerleştirin.***

Not: ***'0x' ön eki bir fark oluşturmaz. Aynı şekilde 'PRINT' değeri '0xPRINT' olarak da yazılabilir. Dilediğiniz formatı kullanabilirsiniz.***

Tek dikkat edilmesi gereken KeyID formatının parmak izine kıyasla daha az haneye sahip olması ki bu, tekrar edebileceği anlamına geliyor. Gösterdiğim long-KeyID gibi aynı zamanda bir short-KeyID (Bakmak için bu sefer listelerken "--keyid-format=short" parametresini ekleyebilirsiniz) formatı bulunuyor ve bunun tekrar edebileceği kanıtlandı. Yani çok büyük bir şanssızlıkla aynı long-KeyID'ye sahip anahtarlarınız olabilir. Ha, aynı şekilde kozmik değerlerdeki bir ihtimalle yine şanssız çıkarsanız parmak izlerinin bile aynı olması mümkün. Ancak ihtimallerin bu boyutunu olası kabul edecek isek kafa yoracak daha büyük sorunlarımız var. Rahat etmek için parmak izini kullanmayı tercih edin.

2-c-2. *Kendi Anahtarınızı oluşturun*

Bizim için dosya şifrelenmesi ve şifrenenleri çözebilmek, anahtarları güvenilir olarak işaretlemek, imzalamak ve bizim için imzalananları doğrulamak, kendi mesajlarımızı imzalamak ve başkaları için dosya şifrelemek gibi işlemler için kendimize bir Ana ve dahilinde bir Alt anahtar oluşturmamız gerekiyor.

Oluşturmadan önce gizli anahtarınız için bir parola ve kullanım dışı kalacağı bir tarih belirleyin. Sonra **EN AZ BİR TANE**sini seçin kullanmak üzere isim, E-Mail ve ek açıklama/yorum belirleyin. Bunlar ayırt etmede işe yarayacaktır.

Şimdi bir Uçbirim/Komut İstemi penceresi açın ve şu komutu parametreleri ile beraber çalıştırın:

```
gpg --full-generate-key
```

Çıkan yanıt isteminde birlikte ilerleyelim:

```
Please select what kind of key you want:
   (1) RSA and RSA
   (2) DSA and ElGamal
   (3) DSA (sign only)
   (4) RSA (sign only)
   (9) ECC (sign and encrypt)
  (10) ECC (sign only)
  (14) Existing key from card
Your selection?
```

RSA and RSA	Alt ve Gizli/Açık Anahtarda RSA kullan. Şifreleme ve İmzalama yapılabilir.
DSA and ElGamal	Gizli/Açık Anahtarda DSA, Alt Anahtarında Elgamal kullan. Şifreleme ve İmzalama yapılabilir.
DSA (sign only)	Gizli/Açık Anahtarda DSA kullan. Alt anahtar oluşturmaz, dolayısıyla sadece imzalamak için kullanılabilir.
RSA (sign only)	RSA kullanan Gizli/Açık Anahtar oluştur. Alt anahtar oluşturmaz, dolayısıyla sadece imzalamak için kullanılabilir.
ECC (sign and encrypt)	Gizli/Açık ve Alt Anahtarda ECC (Eliptik Eğri Kriptografisi) kullan. Şifreleme ve İmzalama Yapılabilir.
ECC (sign only)	Gizli/Açık Anahtarda ECC (Eliptik Eğri Kriptografisi) kullan. Alt anahtar oluşturmaz, dolayısıyla sadece imzalamak için kullanılabilir.
Existing key from card	(Mevcutsa) Akıllı kartta bulunan anahtarı kullan. Şifreleme ve İmzalama yeteneği içindeki anahtara bağlıdır.

Pekala, hangi kriptografi sistemini kullanmalıyım? Bilmeniz gereken şunlar:

ElGamal	Benzer güvenlikte düşük performans
DSA	OpenSSH 2025'te desteğini kesmeyi planlıyor
RSA	Popüler
ECC	RSA'den daha hızlı ama yeni olduğu için yaygın değil

İletişim kurabilmek için karşınızdakiyle uyumlu sistemler kullanmanız gerektiğini unutmayın.

ECC daha yeni olsa bile popülerliği ve güvenilirliğinden ötürü RSA benim için en iyi seçimdir. (Daha ileri seviye kullanımda ayrı alt anahtarlarda ECC kullanmanız tavsiye edilir)

Şifreleme ve İmzalama yeteneği olan bir anahtar istediğim için 1. seçeneği seçiyorum.

(1 yazıp Enter tuşuna basıyorum)

EK 1: *Bu kriptografik sistemleri tanıyorsanız ve parametrelerinde ince ayar yapmak istiyorsanız yukarıdaki komuta "--expert" parametresini ekleyebilirsiniz*

```
gpg --expert --full-generate-key
```

2-c-2. *Kendi Ana Anahtarınızı oluşturun*

Not: *ECC'de algoritmada tanımlı, bu istemle karşılaşmayacaksınız.*

```
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072)
```

Sırada anahtarın bit uzunluğunu seçiyoruz.

2048 Bit minumumdur. Şu anki teknoloji ve devasa bir bütçe ile birkaç ay-yıl içinde kırılması mümkün. Ancak mümkün olması olacağı anlamına gelmiyor. Bunu yapmak isteyen hedeflerine farklı yollardan daha az maliyetle ulaşabilir. Temelinizi sağlam tuttuktan sonra kaygılanın.

Ben seçmekte özgür olduğum için 4096 **yazıp Enter'a basıyorum**

Şimdi anahtarın geçerli olacağı süreyi seçmeliyiz. 0, yani hep geçerli olsun seçeneğini tavsiye etmem. Feshetme sertifikamızı yedeklemeyi unutursak ve olur da biri anahtarımızın kontrolünü ele geçirecek olursa yanarız. Anonim değilse sorun olmayabilir tabi ki.

Ben ideal bulduğum 8 yılı ömrü tanımayı seçiyorum. Bunun için 8y **yazıp Enter tuşuna basıyorum.**

```
Requested keysize is 4096 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0)
```

```
Key is valid for? (0) 8y
Key expires at 10/11/32 13:38:22 Turkey Standard Time
Is this correct? (y/N)
```

Bize ne zaman geçersiz kılınacağını tarih ve zaman dilimi bazında saat olarak belirtiyor. Ve bizden onay bekliyor.

Onaylamak için y **yazıp Enter tuşuna basıyorum**

Teknik kısmı şimdilik bitti. Sırada anahtarımıza isim, e-Mail ve ek açıklama/yorum gibi tanımlayıcılar koyacağız.

Öncelikle isim isteniyor. Bill Gates **yazıp Enter'a basıyorum**

```
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name:
```

Yanıt istemi kendini verdiğim girdiyle güncelledi. Şimdi bir e-Mail adresi istiyor. billgates@microsoft.com **yazıp Enter'a basıyorum.**

```
Real name: Bill Gates
Email address:
```

Yanıt istemi gördüğünüz gibi kendini güncelledi. Şimdi bir ek açıklama/yorum istiyor. Kılavuzumda göstermelik **yazıyorum ve Enter'a basıyorum**

```
Real name: Bill Gates
Email address: billgates@microsoft.com
Comment:
```

2-c-2. *Kendi Ana Anahtarınızı oluşturun*

```
Real name: Bill Gates
Email address: billgates@microsoft.com
Comment: Kılavuzumda göstermelik
You are using the 'utf-8' character set.
You selected this USER-ID:
    "Bill Gates (Kılavuzumda göstermelik) <billgates@microsoft.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit?
```

Bilgilerin doğruluğunu kontrol etmem isteniyor.
Onaylamak için O**,**
e-Mail kısmını düzenlemek için E**,**
Yorum/Ek açıklama kısmını düzenlemek için C**,**
İsim kısmını düzenlemek için N**,**
Çıkmak için Q **yazıp Enter'a basın.**

Ben ek açıklama/yorum kısmını düzenlemek istiyorum o yüzden C **yazıp Enter'a basıyorum.**

Silmek için boş bırakıyorum ve Enter'a basıyorum.

```
Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? C
Comment:
```

```
Comment:
You selected this USER-ID:
    "Bill Gates <billgates@microsoft.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit?
```

Tam istediğim gibi oldu.
Onaylamak için O **yazıp Enter'a basıyorum**

Şimdi anahtarımız için belirlediğimiz parolayı giriyoruz.

Parolayı girdikten sonra TAB tuşunu kullanarak alttaki menüde seçim yapıyoruz. Devam etmek için işaretçiyi <OK>'in üzerine getirin ve Enter'a basın

```
Please enter the passphrase to
protect your new key

Passphrase: ________________________________________

      <OK>                                <Cancel>
```

```
Please re-enter this passphrase

Passphrase: ________________________________________

      <OK>                                <Cancel>
```

Doğrulamak için tekrardan parolayı girmemiz gerekiyor.

Aynı şekilde girdikten sonra TAB tuşuna basarak işaretçiyi oynatıyoruz ve seçim yapabiliyoruz.

Devam etmek için işaretçiyi <OK>'in üzerine getirin ve Enter'a basın

2-c-2. *Kendi Ana Anahtarınızı oluşturun*

```
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
```

Anahtar oluşturulurken bilgisayarınızı kullanmaya devam etmeniz rastgeleliğini, dolayısıyla güvenliğini arttır.

Veya bitmesini de bekleyebilirsiniz.

Çıktıda bana feshetme sertifikamın nerede oluşturulduğu ve anahtarlarımın oluşturulup imzalandığını söylüyor.

```
gpg: revocation certificate stored as
'/home/anon/.gnupg/openpgp-revocs.d/50512F18EAB084D1B0AAA396B0FACF7FF670C5E0.rev'
public and secret key created and signed.
```

Yeni anahtarımın bilgileri de bu şekilde görüntüleniyor.
Bu bilgilerin ne anlama geldiğini öğrenmek için 9. sayfaya dönün.

```
pub   rsa4096 2024-10-13 [SC] [expires: 2032-10-11]
      50512F18EAB084D1B0AAA396B0FACF7FF670C5E0
uid                      Bill Gates <billgates@microsoft.com>
sub   rsa4096 2024-10-13 [E] [expires: 2032-10-11]
```

Pekala, bu “Feshetme Sertifikası” ne?

2-c-2-a. *Feshetme Sertifikası*

Anahtarınızın artık geçersiz/kullanım dışı olduğunu duyurmak için yayınladığınız, sizden geldiği doğrulanabilen sertifikaya “feshetme sertifikası” denir.

Kullanıcılar aynen bir açık anahtar gibi (İçeriğini okursanız bir açık anahtar ile aynı formatta oluşturulduğunu görebilirsiniz.) **içe aktarabilir ve aktardıklarında açık anahtarınız “feshedildi” olarak işaretlenir. Bu sertifikayı bir keyserver’a yüklerseniz aynı şekilde orada da aynı şekilde işaretlenir.** Feshedilene kadar imzalanmış veriler halen feshedilmiş bir anahtar ile doğrulanabilir ancak feshedilmiş bir açık anahtar için yeni veri şifrelenemez. Bu sertifikayı yedeklemeyi ve gizli tutmayı unutmayın.

Feshetme sertifikasını kaybettiyseniz ve halen gizli anahtarınıza erişiminiz varsa & anahtarınızın parmak izi değerine ‘PRINT’, çıktı dosyasının ismine ‘sertifika’ dersek (o an bulunduğunuz dizine kaydedilir) şu komut ve parametreleri ile tekrardan oluşturabilirsiniz:

```
gpg --output sertifika.asc --generate-revocation PRINT
```

2-c-2-b. *Gizli anahtarlarınızı listeleyin*

Oluşturduğunuz gizli anahtarlarınızı görüntülemek için şu komutu parametreleri ile çalıştırın:

```
gpg --list-secret-keys
```

```
sec   rsa4096 2024-10-13 [SC] [expires: 2032-10-11]
      50512F18EAB084D1B0AAA396B0FACF7FF670C5E0
uid           [ultimate] Bill Gates <billgates@microsoft.com>
ssb   rsa4096 2024-10-13 [E] [expires: 2032-10-11]
```

Açık anahtarlardakine benzer bir çıktıyla karşılaşıyoruz.

Buradaki fark ilk satırdaki “pub**” (Açık Anahtar) ifadesi yerine “**sec**” (Gizli Anahtar) gelmesi ve 4. satırdaki “**sub**” (Alt Anahtar) yerine “**ssb**” (Gizli Alt Anahtar) gelmiş.**

Aynı zamanda kendi anahtarımız olduğu için nihai geçerliliği [ultimate] **olduğunu görebilirsiniz.**

Not: ***9. sayfada*** *gösterilen ek parametreleri burada da kullanabilirsiniz.*

2-c-2-c. *Gizli anahtarınızı silme*

Açık anahtar silmeye çok benzer bir işlem, sadece parametrede küçük bir değişiklik yapacağız.

Gizli anahtarımızın parmak izine ‘S_PRINT’, KeyIDsini tercih ederseniz ona da ‘S_KEYID’ diyelim. Gizli anahtarımızı silmek için şu komutu ve parametreleri size uygun şekilde değiştirerek çalıştırın:

```
gpg --delete-secret-key S_PRINT
```

veya

```
gpg --delete-secret-key S_KEYID
```

Emin olduğunuzu doğrulamak için art arda “silmek istediğinize EMİN MİSİNİZ?” uyarılı, arayüzüne artık aşina olduğunuz 4 yanıt istemi ile karşılaşacaksınız.

Silmek için her birini onaylayın.

Not: Gizli *anahtarınız silinmesi açık anahtarınızı yok etmiyor. İşinize yaramayacağından onu da ayrıca* ***10. sayfada*** *gösterildiği gibi silebilirsiniz.*

2-c-3. *Bir anahtarı imzalayın*

İçeriye aktardığınız bir anahtarı doğrudan şifreleme için kullanmanız mümkün, ancak veri doğrulayabilmek için bu anahtarı *onaylamak* yani imzalamak gerekiyor. En basiti olan yerel imzalamayı göstereceğim.

Öncelikle imzalamak için kullanacağınız gizli anahtarı seçin. Bu anahtarın parmak izine 'S_PRINT' diyelim. İmzalayacağınız açık anahtarın parmak izine de 'PRINT' diyelim.

Anahtarı imzalamak için seçimlerinize uygun şekilde değiştirerek parametreleri ile şu komutu çalıştırın:

```
gpg --local-user S_PRINT --lsign-key PRINT
```

Not: ***Parmak izi yerine KeyID'yi kullanabilirsiniz.***
Gizli anahtarın KeyIDsine 'S_KEYID', açık anahtarım KeyIDsine 'KEYID' dersek şu komut ve parametreleri çalıştırarak aynı sonucu elde edebiliriz.

```
gpg --local-user S_KEYID --lsign-key KEYID
```

***KeyID** hakkında daha fazla bilgi için **9. ve 10. sayfayı** kontrol edin.*

```
pub  rsa4096/9C4409899EA8DA7E
     created: 2024-10-14  expires: 2032-10-12  usage: SC
     trust: unknown       validity: unknown
sub  rsa4096/C4C174D867438789
     created: 2024-10-14  expires: 2032-10-12  usage: E
[ unknown] (1). Anon_TestKey (Pratik için hazırlanan test anahtarı)
```

Bu aynı "uid" altında toplanmış imzalayacağımız anahtarların genel görüntüsü.

Not: ***Ayrı ayrı "uid"ler var ise teker teker im zalayabiliyorsunuz.***

```
pub  rsa4096/9C4409899EA8DA7E
     created: 2024-10-14  expires: 2032-10-12  usage: SC
     trust: unknown       validity: unknown
 Primary key fingerprint: 5743 8AB4 880D A775 D237  5A5D 9C44 0989 9EA8 DA7E

     Anon_TestKey (Pratik için hazırlanan test anahtarı)
```

Spesifik olarak imzalayacağımız anahtar. İmza yeteneği bir tek ana anahtarda olduğu için o belirtilmiş

Not: ***İmzalama yeteneği olan alt anahtar oluşturursanız ise onu imzalayabilirsiniz.***

```
This key is due to expire on 2032-10-12.
Are you sure that you want to sign this key with your
key "Bill Gates <billgates@microsoft.com>" (B0FACF7FF670C5E0)

Really sign? (y/N)
```

İmzalayacağım anahtarın geçersiz kılınacağı zamanı belirtiyor ve seçtiğim anahtarımla imzalamak isteyip istemediğimi soruyor.

İmzalamak için y yazıp Enter'a basıyorum

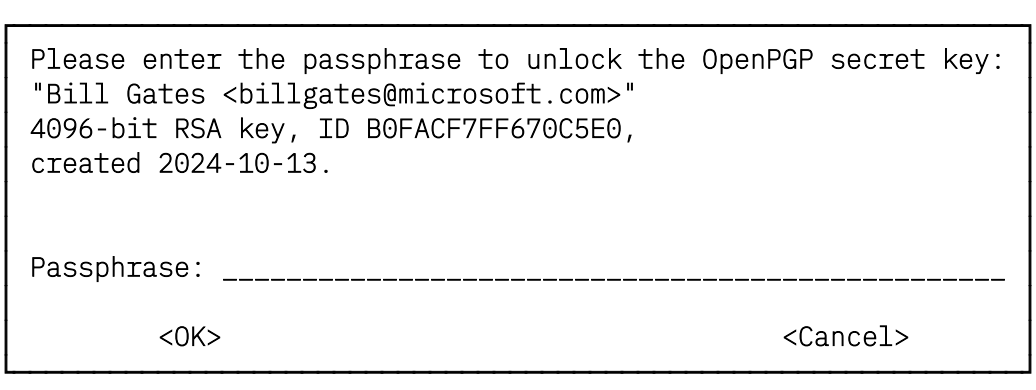
2-c-3. *Bir anahtarı imzalayın*

Şimdi imzalamak için kullandığım gizli anahtarımın parolasını girmem gerekiyor.

Bu ekranda da öncekiler gibi TAB ile seçim yapıyoruz.

Parolamı girip <OK>'i seçiyorum ve Enter'a basıyorum.

Artık benim imzamı taşıyor. Bu durumda artık anahtara TAM güvenilirlik vermiş oldum. Bu noktada anahtarı "yerel" imzaladık. O halde diğer türlüsü de olmalı. Şimdi sıradan imzalamayı öğrenelim.



2-c-3-a. *Bir anahtarı imzalamakla yerel imzalamanın farkı*

Aradaki farkı merak ediyor olabilirsiniz. Kısaca söyle açıklayayım:
Siz herhangi imzalama gerçekleştirdiğinizde anahtarı "onaylamış" olursunuz. **Yerel imzalamada bu "onaylama" sadece kendi anahtarlığınızda geçerlidir çünkü dışarı aktarılamaz**

Sıradan imzalamada ise bu anahtara verdiğiniz "onay" dışarı aktardığınızda birlikte taşınır. Mesala verdiği onayın çok önemli olduğu biri olduğunuzu düşünelim. İnsanlar aldıkları anahtarlarda imzanızı görünce doğruluğuna inanıyorlar. Bu durumda onayınızı göstermek için öncelikle size gönderilen anahtara sıradan imzalama yaparsınız. Sonra bu anahtarı dışarı aktarır ve anahtarını gönderen kişi için şifreleyip mail olarak geri yollarsınız. Böylelikle artık size anahtarı gönderen kişinin halen anahtarının kontrolünde ve mailin kontrolünde olduğunu doğrulayıp anahtarı onayladığınızı karşıdakine iletmiş olursunuz.

Bu noktada anahtarı gönderen maili alır, mesajın şifresini çözer ve imzalanmış anahtarı içeri aktarır. Artık anahtarını tekrar paylaştığında sizin "onay"ınız ile işaretlenmiş olarak gözükür.

Sıradan imzalama yapmak için parametrenizde çok küçük bir değişiklilik yapmak gerekiyor, o da şöyle:

```
gpg --local-user S_PRINT --sign-key PRINT
```

veya

```
gpg --local-user S_KEYID --sign-key KEYID
```

EK 1: *İmzalama sürecinizde yaptığınız araştırmaya ne kadar güvenilebileceğinin bilinmesini istiyorsanız "--ask-cert-level" parametresini inceleyebilirsiniz.*

Referans için bu bağlantıyı kontrol edebilirsiniz

EK 2: Sırasıyla üzerindeki imzaların geçerliliğini kontrol etmek ve görüntülemek veya sadece görüntülemek için şu komutları parametreleri ile çalıştırın:
gpg --check-signatures PRINT **veya** gpg --list-signatures PRINT

2-c-4. *Tanımlayıcılar üzerine*

Elinizdeki herhangi bir tanımlayıcıyı/tanımlayıcıları kullanarak; parmak izi için 'PRINT', KeyID için 'KEYID', UID içinde olan ek açıklama/yorum/e-Mail/İsim gibi şeyleri ayrı ayrı veya bir bütün olarak 'UID' işlem yapabiliyoruz.

Mesala ben anahtarımı 6 farklı şekilde silebilirim:

1. Parmak izi 'PRINT'

gpg --delete-key 50512F18EAB084D1B0AAA396B0FACF7FF670C5E0

veya

gpg --delete-key '5051 2F18 EAB0 84D1 B0AA A396 B0FA CF7F F670 C5E0'

2. KeyID 'KEYID'

gpg --delete-key B0FACF7FF670C5E0

3. UID

gpg –delete-key 'Bill Gates'

veya

gpg --delete-key billgates@microsoft.com

veya ek açıklama/yorumu silmeseyim şu şekilde yapabilirdim:

gpg --delete-key '(Kılavuzumda göstermelik)'

2-c-5. *Kısaca anahtar düzenleme işlemleri*

Anahtarlar üzerinde daha fazla ince ayar yapmak mümkün. Temelde yardım menüsünde görüntülenen, bir nevi sık kullanılan seçenekleri ile bile ileri seviye kullanıcılara hitap ettiği için kısaca bahsedip geçeceğim.

Öncelikle bu menüye erişebilmek için anahtarınızın herhangi bir tanımlayıcısını kullanabilirsiniz. Bu tanımlayıcıya 'ID' diyelim. Tanımlayıcıları tekrardan gözden geçirmek için 18. sayfaya dönebilirsiniz.

Düzenleme menüsüne erişmek için şu komutu uygun parametreleri ile beraber çalıştırın:

gpg --edit-key ID

EK 1: ***Menüyü daha tanıtmadım ama söylemekte fayda var, duraksamadan komut arayüzünde erişmek yapmak istediğinizi art arda yukarıdaki komutun sonuna yazabilirsiniz. Mesala arayüze eriştiğinizde gerçekleştirdiğiniz işlemlere sırasıyla 'a1', 'a2' ve 'a3' diyelim bunu tek seferde şu şekilde gerçekleştirebilirsiniz:***

gpg --edit-key ID a1 a2 a3

```
anon@hostname:~$ gpg --edit-key
50512F18EAB084D1B0AAA396B0FACF7FF670C5E0
gpg (GnuPG) 2.2.40; Copyright (C) 2022 g10 Code GmbH
This is free software: you are free to change and
redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Secret key is available.

sec  rsa4096/B0FACF7FF670C5E0
     created: 2024-10-13  expires: 2032-10-11  usage: SC
     trust: ultimate      validity: ultimate
ssb  rsa4096/BDB6A25CDF905989
     created: 2024-10-13  expires: 2032-10-11  usage: E
[ultimate] (1). Bill Gates <billgates@microsoft.com>

gpg>
```

gpg'nin komut arayüzüne hoşgeldiniz.

Geçerli komutları görüntülemek için help yazıp Enter'a basın

2-c-5. *Kısaca anahtar düzenleme işlemleri*

```
gpg> help
quit        quit this menu
save        save and quit
help        show this help
fpr         show key fingerprint
grip        show the keygrip
list        list key and user IDs
uid         select user ID N
key         select subkey N
check       check signatures
sign        sign selected user IDs [* see below for related commands]
lsign       sign selected user IDs locally
tsign       sign selected user IDs with a trust signature
nrsign      sign selected user IDs with a non-revocable signature
adduid      add a user ID
addphoto    add a photo ID
deluid      delete selected user IDs
addkey      add a subkey
addcardkey  add a key to a smartcard
keytocard   move a key to a smartcard
bkuptocard  move a backup key to a smartcard
delkey      delete selected subkeys
addrevoker  add a revocation key
delsig      delete signatures from the selected user IDs
expire      change the expiration date for the key or selected subkeys
primary     flag the selected user ID as primary
pref        list preferences (expert)
showpref    list preferences (verbose)
setpref     set preference list for the selected user IDs
keyserver   set the preferred keyserver URL for the selected user IDs
notation    set a notation for the selected user IDs
passwd      change the passphrase
trust       change the ownertrust
revsig      revoke signatures on the selected user IDs
revuid      revoke selected user IDs
revkey      revoke key or selected subkeys
enable      enable key
disable     disable key
showphoto   show selected photo IDs
clean       compact unusable user IDs and remove unusable signatures
from key
minimize    compact unusable user IDs and remove all signatures from
key

* The 'sign' command may be prefixed with an 'l' for local signatures
(lsign),
  a 't' for trust signatures (tsign), an 'nr' for non-revocable
signatures
  (nrsign), or any combination thereof (ltsign, tnrsign, etc.).

gpg>
```

Epey bir şeyler sıraladı, kısaca üzerinden geçmeyi planlıyorum zıya bazıları apayrı bir tavşan deliğine açılıyor ki daha burada görüntülenmeyen birkaç komut biliyorum. Aynı zamanda parametre olarak karşılıkları var. Hatta ana anahtar karşılığını düzenlemek için (bildiğim kadarıyla) bir gereklilik.

help – Bu çıktıyı gösteriyor
fpr – Anahtar parmak izini gösteriyor

grip – Anahtarın keygrip'ini gösteriyor. Keygrip olayı GPG mailinglistinde kelimesi kelimesine şöyle açıklanmış "Protokolden bağımsız olarak açık anahtarı tanımlamanın bir yoludur."

uid – Numaralandırılmış UID'ler arasından birini seçmek için, örnek olarak "uid 1" yazıp Enter'a basarsam
"(1). Bill Gates <billgates@microsoft.com>"i seçer.

key – Numararalandırılmış alt anahtarlar arasından seçmek için, örnek olarak "key 1" dersem "ssb"yi seçer

check – Üzerindeki imzaların geçerliliğini kontrol eder
tsign – Anahtarınızın imza yetkisini güvendiğiniz bir başkası ile paylaşmak isterseniz adınıza başkalarının anahtarını imzalaması için güvendiğinizin anahtarını bu "Güven İmzası"ndan yapıyorsunuz. Pek kullanılmaz.

nrsign – Kaldırılamayan/Geçeriz kılınamayan bir imza çeşidi bu. Anahtarınızı kaybetseniz ve feshetseniz bile bu geçerli olmaya devam ediyor o yüzden **HİÇ ÖNERMEM**.

addphoto – İsminden de anlaşılacağı gibi bir fotoğraf ekliyor. Yakaya takılan güvenlik anahtarlarındaki gibi. Ama anahtarınızı çok büyük yapar ve paylaşırken sıkıntı yaşayabilirsiniz. Yapacaksanız düşük çözünürlük seçin.

notation - Spesifik bir imza gösterimi ayarlamak için

trust – "Güven". Bunun 6 farklı seçeneğini açıklayayım:

> **Ultimate (Nihai):** Anahtar tek başına güven sağlıyor, tek başına doğrulamak için yeterlidir. Oluşturduğunuz anahtarlara varsayılan olarak bu güven seviyesi atanır.
>
> **Full (Tam):** İmzasına güvendiğiniz anahtarlara verilir, mesala arkadaşınızın anahtarına bu güven seviyesini tanımlarsanız onun imzaladığı anahtar içeri aktarılınca tam güvenilir olarak tanınır.
>
> **Marginal (Kısmi):** Marginal güvenilirliği olan 3 anahtar ancak birleşip tam güvenilirlik tanımlayabilirler.
>
> **Unknown:** Varsayılan olarak atanan seçenek, herhangi bir güven belirtmiyor.
>
> **Undefined (Belirsiz/Tanımlanmamış):** "Daha karar veremedim" seçeneği
>
> **Never:** "ASLA güvenme". Önüne gelen anahtarı imzalayan birine bunu tanımlayabilirsiniz.

EK 1: *Güven tanımlaması yapmak istediğiniz bir sürü anahtarınız varsa şu komutu parametreleri ile çalıştırarak düzenleyeceğiniz anahtarı değiştirmeyi otomatikleştirebilirsiniz:*

```
gpg --update-trustdb
```

Not: *Bir anahtarı imzaladığınızda "Full Trust" tanımış olursunuz.*

3. *Doğrulama/Şifreleme*

----3-a. *Doğrulama* ----------

3-a-1. *Açık İmza Doğrulama*

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA512

Toplu Parlayanzenci Ölümü --Bill Gates
-----BEGIN PGP SIGNATURE-----

iQIzBAEBCgAdFiEEUFEvGOqwhNGwqqOWsPrPf/ZwxeAFAmcbwSgACgkQsPrPf/Zw
xeC16w/8C2gMSUQ/nrsEPtxo9lYmACBYORjW144fXLKJZw31FwDtiKBEuC9MAq1K
SbGAKzj2OkOW2mgHce9pNlz6bZfwTADf82lOmX//skbdWIBG7XlSdnaYtBuVtF1M
yns5YeGUSh1ruYAbEa7Rrw/FZB89Q0ca76GaVPCInzMZWjeIrMk/ynGIow/vuAxB
YbeIIhvRHMiJNF0kMv7MLmtsOVDFS/4mgi9AezwW/czcKwu64Cxyn970a65JOT6w
Be4BspLZkMTLQED2L+L/t5gsJr7LKmxCLdrsDKz6e3tAuidLMpbD5VG/ke5Atmz6
LCUuigGDkCnSDZMoIs2OAy9p24l0CE4ZRqnGm52dI73WotjiK0JWE1qlG6amA5Fb
GVAB6FwMhqdxN+MsC6sAw9ujYj8Au5cAd6OSUrjX9T8SWPkxWwr9zslQhp2JGHgC
lRzfUgmtPrqfM9UlBT9x8pHAzY4WHj8CeollVRXzOH8ktbhGjCahHIAeMC9iCT+Q
RrYf7GpK3sjxt0z5m5ff6dMraSjQp4Tyx/JTAaCjAgctNkIuoskLjLJV1cALbJRD
ZeGnwkId6h9g1feFRfLTh5ci4GOPlOb34ZuuSLaEFZnZw5GGVYRMGy4p9gLCZgH/
XAiV2WpYUSEVWJgtMfRaH+PhMxGija6K6ZsSeZvoFFT71uSkkYY=
=/OSG
-----END PGP SIGNATURE-----
```

İmzalanan mesaj bu kısımda bulunur

Açık İmzalama yani Clear Sign imzalaması okunabilir ve metin formatında yapıştırılabildiğinden beyan edeceğiniz yazılı metinleri imzalamak için kullanabilirsiniz.

Örnek olarak böyle bir mesaj aldığımızı varsayalım, Bill Gates'in bunu söylediğini doğrulayabilir miyiz?

Not: ***Doğrulamak için 3. sayfadaki açık anahtarı anahtarlınıza eklemeyi unutmayın***

Bir terminal/uçbirim penceresi açın ve şu komutu parametresi ile çalıştırın:

```
gpg --verify
```

Soldaki mesajı yapıştırın (boşluk olması gereken yerlere dikkat) ve Enter'a basın ve sonrasında EOF kombinasyonuna basın.

Not: ***EOF kombinasyonu UNIX için 'CTRL+D', Windows için 'CTRL+Z + Enter'dır.***

Bu girdiyi dosya üzerinden sağlamayı tercih ederseniz bir metin dosyasının içine kaydedin.

Ardından tek yapmanız gereken dosya adını, buna 'FILENAME' diyelim, parametre olarak eklemelisiniz:

```
gpg --verify FILENAME
```

3-a-2. *Açık İmza Doğrulama*

```
gpg: Signature made Fri 25 Oct 2024 07:02:48 PM +03
gpg:                using RSA key 50512F18EAB084D1B0AAA396B0FACF7FF670C5E0
gpg: Good signature from "Bill Gates <billgates@microsoft.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 5051 2F18 EAB0 84D1 B0AA  A396 B0FA CF7F F670 C5E0
```

Mesaj üzerinde oynanmadığını

gpg: Good signature from "Bill Gates <billgates@microsoft.com>"

Satırından anlayabiliyoruz. Ve aynı zamanda anlıyoruz ki Bill Gates bunu gerçekten söylemiş.

3-a-3. *Ayrık İmza Doğrulama*

İçinde verinin kendisini bulundurmadığı için ikisinden birinin kaybolma ihtimali vardır.

Çoğunlukla çıktısı ASCII-Armored alınmışsa ‘.asc’, varsayılan olarak ‘.sig’ şeklinde olur. Ancak unutmayın ki dosya içeriği önemlidir, uzantısı herhangi bir notasyonda olabilir.

Elinizdeki veriye ‘DATA’ diyelim, aldığınız imza ise ‘BSIGN’ olsun. Doğrulamak için şu komutu parametreleri ile çalıştırın:

```
gpg --verify BSIGN.sig DATA
```

veya

```
gpg --verify BSIGN.asc DATA
```

Uçbirim / Komut istemi çıktısında “gpg: Good signature” olması gerekiyor. Aksi halde geçersizdir.

Not: ***İmza, veriden önce girdide bulunmalıdır.***

3-a-3. *Gömülü İmza Doğrulama*

İçinde verinin kendisini de bulunduğu için bütünlük avantajı vardır.

Çoğunlukla çıktısı ASCII-Armored alınmışsa ‘.asc’, varsayılan olarak ‘.gpg’ şeklinde olur. Ancak unutmayın ki dosya içeriği önemlidir, uzantısı herhangi bir notasyonda olabilir.

Aldığınız dosyaya ‘FILE’ diyelim.
Şimdi bir uçbirim / komut istemi penceresi açın ve şu komutu parametreleri ile birlikte yazıp çalıştırın:

```
gpg --decrypt FILE
```

EK 1: ***Çıktıyı yönlendirmek / kaydetmek isterseniz ‘--output’ parametresini ekleyebilirsiniz. İstediğiniz konum ve dosya ismi bütününe ‘O_FILE’ diyelim. Şu komutu parametreleri ile çalıştırarak bunu yapabilirsiniz:***

```
gpg --output O_FILE --decrypt FILE
```

Uçbirim / Komut istemi çıktısında “gpg: Good signature**” olması gerekiyor. Aksi halde geçersizdir.**

----3-b. *Şifreleme* ----------

Öncelikle gönderinizi teslim alacak birinin olması gerekiyor, bu şahsın açık anahtarının parmak izine ‘R_PRINT’ diyelim.

Kullanacağınız gizli anahtarınızın parmak izine de ‘S_PRINT’ diyelim.

3-b-1. *Mesaj şifreleme*

Şu komutu parametreleri ile beraber çalıştırın:

```
gpg --armor --local-user S_PRINT --recipient R_PRINT --encrypt
```

Hazırladığınız mesajı yapıştırın veya baştan yazın, sonrasında Enter’a basın ve platformunuzun EOF kombinasyonuna basın. Çıktıda ‘-----BEGIN PGP MESSAGE-----’ ile başlayıp ‘-----END PGP MESSAGE-----’ ile biten kısmı bu belirteçleri ile beraber kopyalayın ve alıcıya gönderin.

Mesajımız güvende, artık herkese açık iletişim kanallarında paylaşılabilir. İletiyi belirlenen alıcı okuyabilir.

3-b-2. *Dosya şifreleme*

Şifreleyeceğimiz dosyanın adı ‘FILE’ **olsun.**
Şu komutu parametreleri ile beraber çalıştırarak ‘FILE’ **şifrelenebilir:**

```
gpg --local-user S_PRINT --recipient R_PRINT --encrypt FILE
```

çıktıyı adlandırmak veya kaydedileceği konumu değiştirmek isterseniz, ki bu ad & konum bütününe ‘O_FILE’ **diyelim, bunu** ‘--output’ **parametresini ekleyerek gerçekleştirebilirsiniz:**

```
gpg --local-user S_PRINT --recipient R_PRINT --output O_FILE --encrypt FILE
```

EK 1: *Çıktıyı yapıştırılabilir metin olarak formatlamanız mümkün, ancak girdideki dosyanın boyutuna göre çok uzun bir metin olabilir. Bunu* ‘*--armor*’ *parametresi ekleyerek gerçekleştirebilirsiniz:*

```
gpg --armor --local-user S_PRINT --recipient R_PRINT --encrypt FILE
```

siz belirlemezseniz çıktı dosyası, ‘FILE.asc’ **olacak ve bu ASCII-Armored formatında şifrelenmiş dosyayı içerecektir.**

Simetrik şifreleme kullanmayı tercih ederseniz şu komutu parametreleri ile birlikte çalıştırın:

```
gpg --symmetric FILE
```

Dosya için bir parola belirlemeniz gerekir.

Unutmayın ki simetrik şifrelemede parolayı bilen herkes dosyanın şifresini çözebilir

4. *Anahtar Yayınlama*

Şahsi veya size paylaşılan anahtarların birinin parmak izine ‘A_PRINT’ diyelim

----4-a. *ASCII-Armored Çıktısı* ----------

Kendi anahtarınızın ASCII-Armored formatında açık anahtarını çıkartmak için şu komut ve parametrelerini yazıp çalıştırın:

```
gpg --armor --export A_PRINT
```

Çıktıyı paylaşırken ‘-----BEGIN PGP PUBLIC KEY BLOCK-----’ ve ‘-----END PGP PUBLIC KEY BLOCK-----’ belirteçlerini dahil etmeyi unutmayın.
Dosya içine kaydetmek isterseniz ‘--output’ parametresini ekleyip çalıştırabilirsiniz. Bu dosyaya ‘O_FILE’ der isek şu komutu parametreleri ile çalıştırabilirsiniz:

```
gpg --armor --output O_FILE --export A_PRINT
```

Sırada insanlara duyurmak var.

----4-b. *Bir Keyserver’a gönderin* ----------

Göndereceğiniz Keyserver’ı seçin, buna ‘KEYSERVER’ diyelim. Şu komutu parametreleri ile birlikte çalıştırarak gönderebilirsiniz:

```
gpg --keyserver KEYSERVER --send-keys A_PRINT
```

5. *İmzalama/Şifre Çözme*

----5-a. *İmzalama* ----------

İmzalamak için kullanacağınız şahsi anahtarınızın parmak izine ‘S_PRINT’ diyelim.

5-a-1. *Açık İmza Oluşturma*

Şu komutu parametreleri ile birlikte çalıştırarak açık imza oluşturabilirsiniz:

```
gpg --local-user S_PRINT --clear-sign
```

Şimdi kopyaladağınız metni yapıştırabilir veya baştan yazabilirsiniz. Girdiyi onaylamak için Enter’a basın ve EOF kombinasyonunu gerçekleştirin.

Parolanızı doğruladıktan sonra açık imzalı mesajınız çıktıda belirecektir.

5-a-2. *Ayrık İmza Oluşturma*

İmzalayacağınız dosyanın adına ‘FILE’ diyelim. Şu komutu parametreleri ile birlikte çalıştırarak bunu yapabilirsiniz:

```
gpg --local-user S_PRINT --detach-sign FILE
```

EK 1: *‘--output’ parametresini kullanarak ayrık imza dosyasının kaydedileceği lokasyonu ve dosya adını belirleyebilirsiniz*

EK 2: *‘--armor’ parametresini ekleyerek ayrık imzayı ASCII-Armored formatta kaydedebilirsiniz*

Parolanızı doğruladıktan sonra çıktı bulunduğunuz dizine kaydedilecektir.

5-a-3. Gömülü İmza Oluşturma

İmzalayacağınız dosyanın adına ‘FILE’ diyelim. Şu komutu parametreleri ile birlikte çalıştırarak bunu yapabilirsiniz:

```
gpg --local-user S_PRINT --sign FILE
```

EK 1: *‘--output’ parametresini kullanarak ayrık imza dosyasının kaydedileceği lokasyonu ve dosya adını belirleyebilirsiniz*

EK 2: *‘--armor’ parametresini ekleyerek ayrık imzayı ASCII-Armored formatta kaydedebilirsiniz*

Parolanızı doğruladıktan sonra çıktı bulunduğunuz dizine kaydedilecektir.

----5-b. *Şifre Çözme* ----------

Alıcı şahsi anahtarınızın parmak izine ‘S_PRINT’ diyelim.

Şu komut ve parametreleri ile aldığınız mesajı çözebilirsiniz:

```
gpg --local-user S_PRINT --decrypt
```

Şimdi kopyaladağınız şifrelenmiş mesajı yapıştırabilir veya yazabilirsiniz. Girdiyi onaylamak için Enter’a basın ve EOF kombinasyonunu gerçekleştirin.

Parolanızı doğruladıktan sonra çözülmüş mesajınız çıktıda belirecektir.

Eğer karşı taraftan bir dosya aldıysanız, bu dosyanın ismine ‘FILE’ diyelim, şu şekide şifresini çözebilirsiniz:

```
gpg --local-user S_PRINT --decrypt FILE
```

Parolanızı doğruladıktan sonra çözülmüş dosyanız bulunduğunuz dizine kaydedilecektir. Eğer ki çıktı dosyanın adını veya kaydedileceği lokasyonu değiştirmek isterseniz ‘--output’ parametresini kullanabilirsiniz.

Not: *Simetrik şifreleme çözmek için ‘--local-user’ parametresi belirtmeniz gerekmiyor.*

`gpg --output O_FILE --decrypt FILE` ***yeterlidir. Devamında dosyayı çözmek için parolayı girin.***

Bazı İpuçları

1. Bir veri hem imzalanıp hem de şifrelenebilir. Ancak bu farklı kullanım senaryolarına uygun bir kullanımdır.

2. GPG anahtarlığınızın "ev klasörü" lokasyonu '`--homedir`' parametresi ile belirtilebilir. Örnek olarak pratik yaparken farklı bir kullanıcıyı taklit etmek için kullanılabilir.

3. Uzantılara göre içeriği tahmin edebilirsiniz. '`.key`' anahtarları, '`.sig`' harici imzaları, '`.gpg`' gömülü imzaları veya şifrelenmiş dosyaları (ikisinin de çözülmesi gerekiyor "`--decrypt`") ve '`.asc`' ise ASCII-Armored formatında bulunduğunu belirtiyor.

4. Gizli anahtarlarınızı aynı bir açık anahtar gibi dışarı aktarabilirsiniz. Bunun için '`--export`' yerine '`--export-secret-key`' parametresini kullanın. **ÇIKTIYI SAKLAMAYI UNUTMAYIN.**

5. Anahtarları topluca dışarı aktarmak için '`--export`' ve '`--export-secret-key`' parametrelerinin devamını boş bırakın. Dışarı aktarılacak anahtarı spesifik olarak belirtmezseniz hepsi dışarıya aktarılır.

6. Birden fazla kişinin aynı şifrelenmiş dosyayı çözebilmesini istiyorsanız birden fazla '`--recipients`' parametresi ayarlayın. Her birine farklı anahtar yazmayı unutmayın.

7. Daha fazla ipucu için arama motorlarında "GPG cheat sheet" veya "PGP cheat sheet" araması gerçekleştirebilirsiniz

Sözlük

Ana Anahtar	Master Key
Açık Anahtar	Public Key
Gizli Anahtar	Secret Key
Alt Anahtar	Subkey
Gizli Alt Anahtar	Secret Subkey
Nihai Güven Seviyesi	Ultimate Trust
Tam Güven Seviyesi	Full Trust
Kısmi Güven Seviyesi	Marginal Trust
Bilinmeyen Güven Seviyesi	Unknown Trust
Belirsiz Güven Seviyesi	Undefined Trust
“Asla” Güven Seviyesi	Never Trust
Açık İmza	Clear Signature
Ayrık İmza	Detached Signature