GPG in Linux

Chari Karipidis
2TinG

24 mei 2012

Inhoudsopgave

1 Introductie

In dit document wordt het gebruik van GPG(GnuPG) nader verklaard.

Benadering De geschiedenis wordt in sectie 2.1 benaderd. GPG wordt uitgelegd in Sectie 2.2. Hierin wordt de werking en uitvoer via command-line benaderd in secties 2.3 en 2.4. Ook zijn er Grafische applicaties die met GPG werken. Deze worden benaderd in secties 2.5 en 2.6. In de bijlagen sectie 3 wordt uitgelegd hoe een versiebeheersysteem ingesteld(3.1), een mailserver(3.2) en een script voor een mail te verzenden (3.3) geconfigureerd worden.

2 GPG

2.1 Geschiedenis

Er is altijd wel een probleem met boodschappen verzenden en ontvangen, zonder dat men deze kan onderscheppen en lezen. Hier zijn handige uitvindingen voor ontworpen, die helpen bij dit probleem.

Scytale In de tijd van de romeinen had men een manier nodig om berichten te versturen naar geallieerde troepen. Verzender en ontvanger waren in het bezit van een "Scytale" van ieder dezelfde grootte. Dit voorwerp was een soort van cilinder. Hier werd een riem over gewikkeld en een boodschap opgeschreven.
Bij het verwijderen van de riem was deze tekst onleesbaar zonder behulp van de Scytale. De letters waren namelijk door elkaar. Bij ontvangst van de riem bij de troepen wikkelden ze de riem over de Scytale die zij bezitten en was het zo mogelijk om de boodschap te lezen.
Dit was een soort van encryptie. Ervoor zorgen dat een onderschepper, de boodschap niet kan lezen.

Caesar methode Een andere encryptiemethode was de Caesar methode. Deze bestond uit een zin hervormen m.b.v. het alfabet.
Dit klinkt natuurlijk zeer logisch. Het alfabet wordt namelijk gebruikt om zinnen te schrijven.
Maar na het schrijven van de nodige boodschap, wordt er een "sleutel"gekozen. Deze sleutel is een afgesproken cijfer tussen 1 en 26, tussen beide partijen.
Belangrijk is dat de cijfers overeenkomen met een letter uit het alfabet. Als het gekozen cijfer 6 is, wordt het alfabet 6 maal naar links verschoven. A wordt dan F en B wordt dan G, etc..
De ontvanger krijgt dan een wirwar van letters en kan deze ontcijferen door het alfabet terug te vormen door het 6 maal naar rechts te verschuiven.

Heden Tegenwoordig worden loopjongens niet meer gebruikt. Men is mee geëvolueerd naar de toekomst.
Technologie is nu de heerser over het verzenden van boodschappen. Mailen, accounts aanmaken, bestanden opslaan, etc.. Gebeurt iedere dag. Dit moet dan ook beveiligd worden.
Dit doen we aan de hand van het encrypteren van de bestanden, handtekenen van mails, etc.. Een manier voor encryptie is GPG.

2.2 Wat is GPG?

GPG of GnuPG staat voor: Gnu Privacy Guard. Zoals de naam al voorstelt, is het om de privacy van gebruikers te beschermen. Dit doormiddel van encryptie van boodschappen die verzonden moeten worden zoals mails, data encrypteren, "sleutelhangers", etc..
GnuPG is een commando voor de terminal, zoals te zien in Subsectie 2.3, maar er zijn dergelijke frontend programma's om deze in een grafische applicatie te kunnen gebruiken. Te zien in Subsectie 2.5

2.3 GPG in command-line interface (CLI)

Zoals ieder ander commando, heeft GPG ook zijn nodige syntax.

$$gpg[--homedirname][--optionsfile][options]command[args]$$

Het commando "GPG"heeft een enorm aantal opties. In "man gpg"worden de opties weergegeven met de nodige uitleg.
In subsecties 2.3.1, 2.3.2 en 2.3.3 worden voorbeelden weergegeven van welke er het meest gebruikt worden.

2.3.1 Voorbeelden van vaak gebruikte commando's

[1]	-c	Symmetrische encryptie, vraagt voor passphrase.
	–decrypt	Decryptie van geëncrypteerde bestanden.
	–encrypt	Encryptie van data. Wordt gecombineerd met -sign.
	-sign	Maakt handtekening, wordt gecombineerd met –encrypt.
	–encrypt-files	Encryptie van meerdere bestanden in 1 commando.
	–decrypt-files	Decryptie van meerdere bestanden in 1 commando.

Tabel 1: Vaak gebruikte commando's

2.3.2 Voorbeelden van vaak gebruikte opties

[1]	-o file	Schrijft output naar "file".
	–default-key name	Standaard waarde voor ID encryptie.
	-r name	Encryptie naar ontvanger "name".
	-v	Verbose, geeft meer info tijdens het proces.
	-i	Interactief, geeft prompts voor iedere stap.

Tabel 2: Vaak gebruikte opties

2.3.3 Voorbeelden van het gebruik

[1]	$gpg - rBobfile$	Handteken en encryptie voor Bob.
	$gpg - -clearsignfile$	Maakt een lege handtekening.
	$gpg - -fingerprintuser_ID$	Laat vingerafdruk zien.
	$gpg - -verifygpgfile$	Verifieert pgpfile.
	$gpg - -list - keysuser_ID$	Laat sleutels zien.

Tabel 3: Voorbeelden

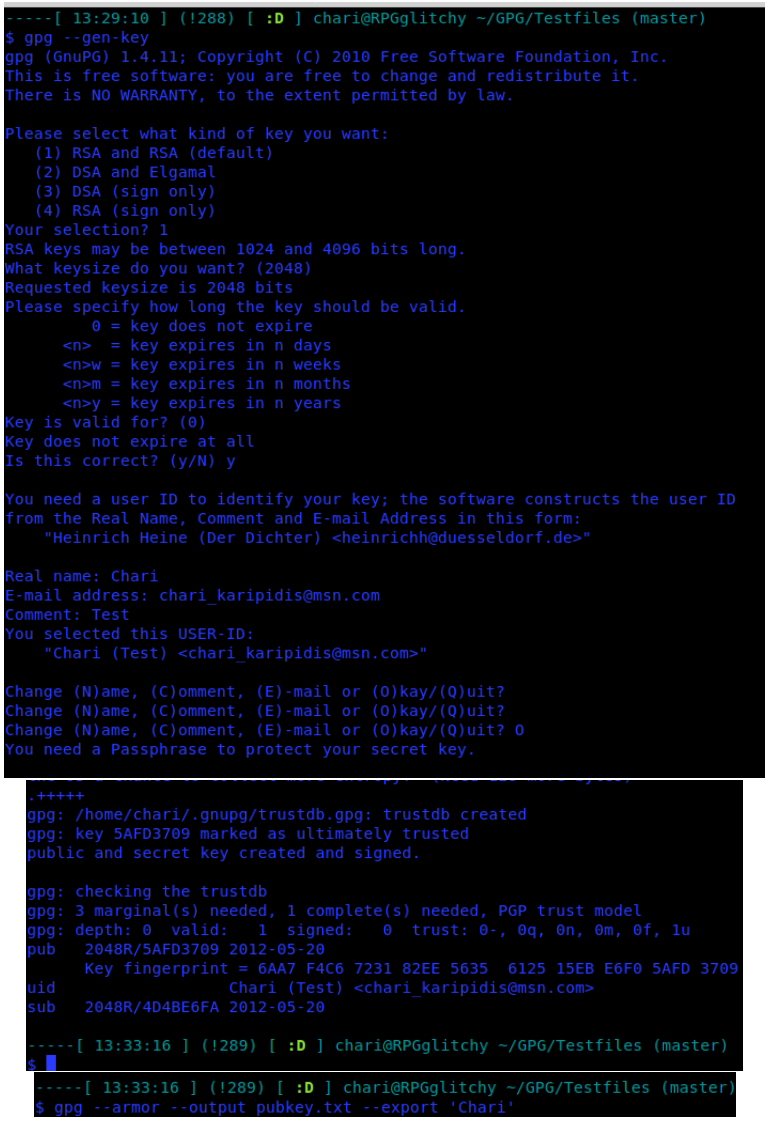
[9]

2.4 Werking en uitvoer van GPG

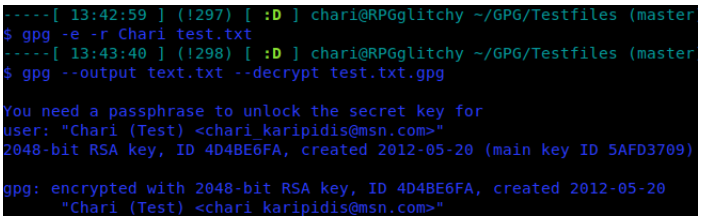
Sleutel Om een bestand te kunnen encrypteren, is een sleutel vereist.
Deze moet als initieele stap worden aangemaakt met het commando:
gpg –gen-key
Bij het gebruik van dit commando worden veel vragen gesteld ter configuratie.
-type van sleutel, -grootte, -vervaldatum, -identificatie.
Na het ingeven van deze waardes wordt er gevraagd naar een Passphrase. Deze is zeer belangrijk te onthouden.
Door het commando:
“gpg –armor –output pubkey.txt –export ’Naam’ ”
te gebruiken, zal een uitvoer gemaakt worden van deze public key naar textbestand, om deze te kunnen verzenden naar de gewenste contacten.
Zie figuur 1

Encryptie en decryptie [3] gpg –encrypt –recipient “Naam”test.txt
Door dit commando te gebruiken, zal het bestand “test.txt”geëncrypteerd worden met de sleutel van *Naam*.
Na het gebruik van dit commando, werd het bestand “test.txt”geëncrypteerd naar “test.txt.gpg”. Als er gewenst wordt, dit bestand te decrypteren, dan gebruiken we het volgende commando:
gpg –output test.txt –decrypt test.txt.gpg
Zie figuur 2

De voorgaande methode is bestemd voor persoonlijk gebruik.
Om te encrypteren naar andere contacten maken we gebruik van het commando “importeren”: gpg –import key.asc
In dit commando geldt dat “key”de sleutel is van de persoon die gecontacteerd moet worden.
Een andere mogelijkheid is deze sleutel zoeken op een publieke website:
gpg –search-keys “myfriend@his.isp.com”–keyserver hkp://subkeys.pgp.net
Bij het encrypteren van een bestand, wordt er gebruik gemaakt van dezelfde commando, als bij persoonlijk gebruik, maar met een verschillende sleutel.
gpg –encrypt –recipient “myfriend@his.isp.net”test.txt
[3]



Figuur 1: Sleutel



Figuur 2: Encryptie

2.5 Frontend GPG programma's

[2]

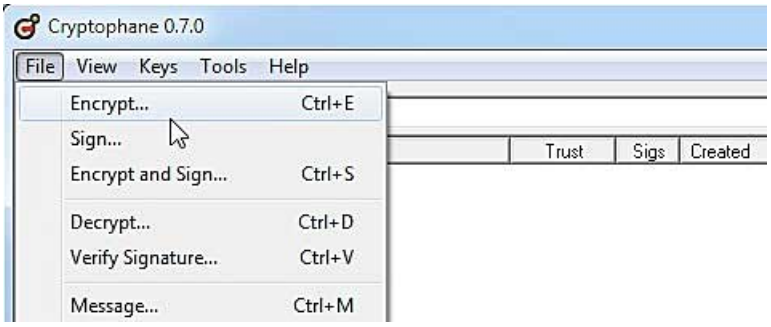
Cryptophane	Een applicatie voor Windows.
Gajim	Een Jabber client voor GNOME.
GnuPG Shell	Een cross-platform, grafische Frontend voor GnuPG.
GPA	De standaard Frontend voor GPG.
KGpg	GnuPG voor KDE.
Seahorse	GnuPG voor GNOME.
Wija	Een cross-platform jabber client (MacOsX, Linux, Windows)

Tabel 4: Gui Frontends

2.6 Werking van een GPG Frontend

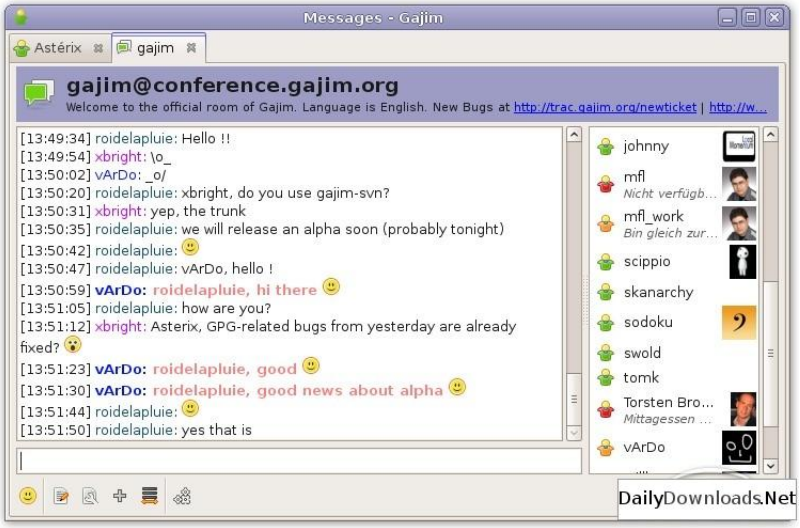
Een grafische applicatie voor het gebruik met GPG houdt heel vaak in dat een bestand geladen wordt en de gebruiker dan kan kiezen om te encrypteren of decrypteren.
Als de Frontend gebruik maakt van een messaging applicatie (Jabber-client), Houdt dit meestal in dat de gebruiker een handtekening instelt in de GUI, zodat deze gebruikt wordt bij het encrypteren van de boodschappen.
Als de GUI een sleutelhanger-functie heeft, zal er een instelling van een "Passphrase"Voorzien zijn bij de initieele stappen. Deze passphrase zorgt ervoor dat de sleutelhanger ontgrendelt kan worden voor gebruik. Als er dan een wachtwoord wordt ingesteld op een website of applicatie, Zal er worden gevraagd deze op te slaan in je sleutelhanger. Als de gebruiker dit toestaat, wordt deze geëncrypteerd toegevoegd aan je sleutelhanger.

Cryptophane Deze wordt gebruikt om te encrypteren, decrypteren, handtekenen, beheer van sleutelhanger en een command-line interface voor GnuPG.



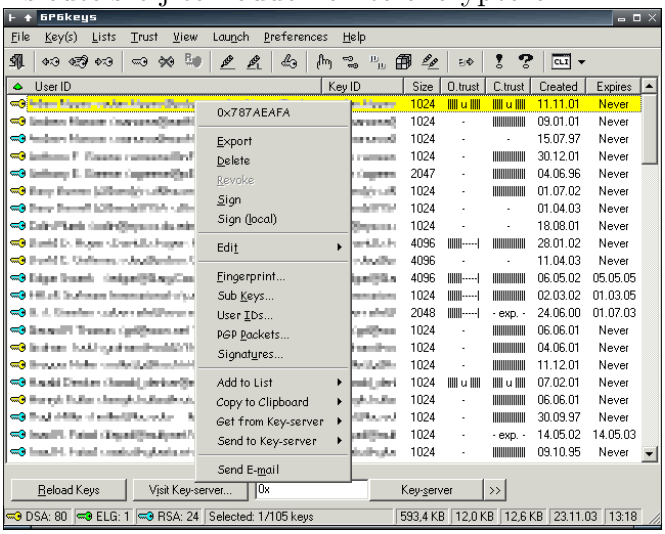
Figuur 3: Cryptophane

Gajim Gajim is een Jabber-client. Een Jabber-client is een messaging applicatie. Omdat Gajim werkt met GnuPG, zullen de berichten die verzonden worden met Gajim, geëncrypteerd worden.



Figuur 4: Gajim

GPGshell [5] Een grafische frontend voor iedere platform. Met deze GUI is het mogelijk sleutels bij te houden en te encrypteren.

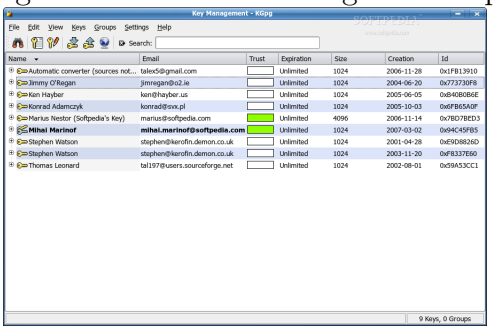


Figuur 5: GnuPG Shell

GPA GPA probeert de standaard frontend te zijn voor GPG. *www.gnupg.org* Host GPA.

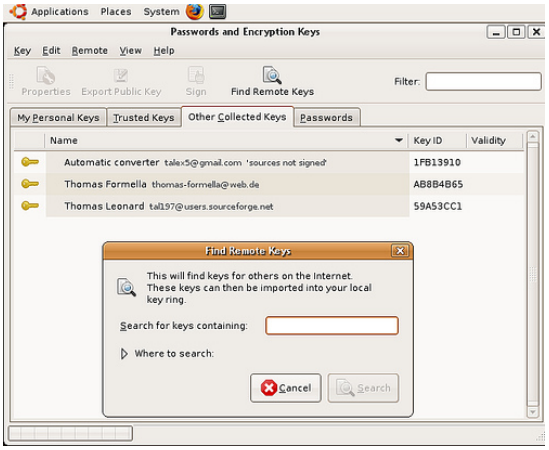
Figuur 6: GPA

KGpg [6] Met KGpg kan je bestanden en mails encrypteren en decrypteren om je informatie veilig te houden. Het is een gratis en open-source frontend.



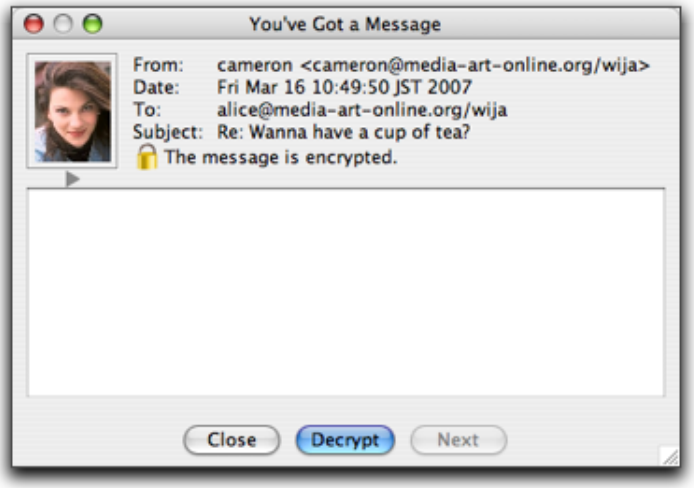
Figuur 7: KGpg

Seahorse [8] Seahorse is een GUI voor GNOME. Het is ook gentegreerd in Nautilus, gedit en andere applicaties voor encryptie uit te voeren. De gebruiker kan met Seahorse PGP en SSH sleutels maken en beheren, publiceren en terughalen van sleutels op de servers, een passphrase opslaan in het cache geheugen, sleutelhanger backuppen, etc..



Figuur 8: Seahorse

Wija Een Jabber-client zoals Gajim, maar geschreven in java en beschikbaar voor ieder platform. Het heeft een ingebouwde sleutelhanger beheersysteem. Het kan ook zeer gemakkelijk boodschappen encrypteren en decrypteren voor gewone gesprekken of multi-user gesprekken. Het is ook mogelijk de boodschappen te handtekenen.



Figuur 9: Wija

Referenties

[1] Anoniem. Commando's. website. http://linux.about.com/library/cmd/blcmdl1_gpg.htm.

[2] Anoniem. Frontends. website. http://www.gnupg.org/related_software/frontends.en.html.

[3] Anoniem. GPG snelstart. website. www.madboa.com/geek/gpg-quickstart/.

[4] Anoniem. GPG snelstart. website. www.Github.com.

[5] Anoniem. GPGShell. website. http://www.jumaros.de/rsoft/index.html.

[6] Anoniem. KGpg. website. http://utils.kde.org/projects/kgpg/.

[7] Anoniem. Mailserver configureren. website. http://shreevatsa.wordpress.com/2007/07/31/using-gmail-with-mutt-the-minimal-way/.

[8] Anoniem. Seahorse. website. http://projects.gnome.org/seahorse/.

[9] Anoniem. Tabellen beheren. website. http://tex.stackexchange.com/questions/8652/what-does-t-and-ht-mean/.

Lijst van figuren

Lijst van tabellen

Index

3 Bijlagen

3.1 Versiebeheersysteem

3.1.1 Registratie

De eerste stap om een versiebeheersysteem te kunnen aanmaken, is het registreren op een website die versiebeheersysteem beheert.
www.github.com Is een beheerder van versiebeheersystemen.
Bij het registreren op Github wordt er nadien een ronde gedaan om te helpen bij het configureren van iemands bewaarplaats.

3.1.2 Configuratie

Via "Synaptic Package Manager"is het nodig, bepaalde pakketen te downloaden, namelijk: "git-core", "git-doc", "git-gui".
Github maakt gebruikt van SSH sleutels voor een veilige connectie.
Met de volgende commando's in de terminal, wordt er een eigen SSH sleutel aangemaakt.
Eerst, bestaande SSH sleutels backuppen en verwijderen.

mkdir key_backup
cp id_rsa* key_backup
rm id_rsa*

Daarna een nieuwe SSH sleutel genereren.

ssh-keygen -t rsa -C "uw_email@uwemail.com"

De terminal vraagt de gebruiker voor een passphrase. Na ingave van de passphrase is de SSH sleutel ingesteld.
Deze sleutel moet toegevoegd worden in github. De volgende stappen laten zien hoe dat gebeurd.
Open id-rsa.pub en kopieer de inhoud van dit bestand.
In de tab "Instellingen"op Github, bij "SSH Keys"is het mogelijk een Sleutel toe te voegen.
Bij het drukken op "Add SSH key"kan de gekopieerde tekst toegevoegd worden en opgeslaan.

Het versiebeheersysteem kan met de SSH sleutel nu getest worden door het volgende commando:

ssh -T git@github.com

Hier zal worden gevraagd om verder te gaan en zal dan de gebruiker begroeten en toegang bieden.

De laatste stap bestaat uit configuratie van git.
git config –global user.name "Voornaam Achternaam"
git config –global user.email "uwemail@uwemail.com"

Deze commando's zorgen voor een correcte configuratie van git.

[4]

3.1.3 Git commando's

Nadat het versiebeheersysteem correct is geconfigureerd, kan er een bewaarplaats worden aangemaakt.
Door op "New repository"te klikken is het mogelijk een bewaarplaats aan te maken in een bestaand account.
Met behulp van volgende commando's, kunnen bestanden in een bepaald pad worden toegevoegd aan de git.

mkdir /Hello-World
cd /Hello-World
git init

Door git init, wordt een connectie geïnitialiseerd tussen github en het pad.
Als er nu bestanden worden toegevoegd in deze map, kan er met de volgende commando's, bestanden toegevoegd worden aan git.

git status (laat zien welke bestanden verandert zijn)
git add *naam (voegt bestand met naam *naam toe)
git commit -m "boodschap voor verandering"
git remote add origin git@github.com:gebruikersnaam/Hello-World.git
git push -u origin master (eerst een origin master aanmaken voor een beheer op afstand)

Nadat er voor het eerst een origin master wordt aangemaakt en "pushed",
Zal er in de toekomst enkel "git push"gebruikt kunnen worden.

3.2 Mailserver configuratie

3.2.1 Mailserver instellen

Om mutt te installeren, wordt er gebruik gemaakt van een simpel commando.

sudo apt-get install openssl mutt

Alle gevraagde waardes mogen standaard zijn.
Mutt is nu succesvol genstalleerd.

3.2.2 Configuratie

Ter configuratie moet een bestand ".muttrc"aangemaakt worden in de home-folder.
Dit bestand moet de volgende inhoud bevatten:

set imap_user = "username@gmail.com"
set imap_pass = "password"

set smtp_url = "smtp://username@smtp.gmail.com:587/"
set smtp_pass = "password"
set from = "username@gmail.com"
set realname = IJour Real Name"

set folder = "imaps://imap.gmail.com:993"
set spoolfile = "+INBOX"
set postponed="+[Gmail]/Drafts"

set header_cache= /.mutt/cache/headers
set message_cachedir= /.mutt/cache/bodies
set certificate_file= /.mutt/certificates

set move = no

In dit bestand moeten de "username", "password"en "Your Real Name"gepersonaliseerd worden.
[7]

3.3 Bash-script

```
#!/bin/bash

echo '                              ____    ____            ____                ____  ____'
echo '|\    /|    /\    |  |     |    |  |      \      / | |    \          / |     |    |'
echo '| \  / |   /  \   |  |     |____|  |____   \    /  | |____ \        /  |____ |____|'
echo '|  \/  |  /----\  |  |     |   \   |        \  /   | |      \  /\  /   |     |   \'
echo '|      | /      \ |  |____ |    \  |____     \/    | |____   \/  \/    |____ |    \'
echo ' '


while true; do
        read -p 'Voorzie een aanspreking: ' aanspreking
        read -p 'De voornaam van de gewenste ontvanger: ' voornaam
        read -p 'De naam van de gewenste ontvanger: ' naam
        read -p 'Het e-mail adress van de persoon die u wents te contacteren: ' email
        read -p 'Geef nu het pad op, waar het .tex bestand zich bevindt: ' pad

        echo 'Zijn de gegevens correct?'
        echo '========================='
        echo $aanspreking $naam $voornaam
        echo 'Het e-mail adres: ' $email
        echo 'Het pad naar het .tex bestand: ' $pad
        echo '========================='
        read -p '[J/N]' correct

        if [[ "$correct" == "J" ] -o [ "$correct" == "j" ]]
                then break;
        fi
done

echo 'Om fouten te vermijden wordt er een backup gemaakt van de huidige .tex file.'
cp "$pad/GPG.tex" "$pad/GPGbackup.tex"

sed -r "s/\rfoot{}/\rfoot{$naam $voornaam }/" "$pad/GPG.tex" > "$pad/GPG.txt" && mv "$pad/GPG.tx

zip -r "$pad/KaripidisChari_$naam$voornaam.zip" $pad > "$pad/zip.txt"

read -p 'Er opent, bij het drukken op enter, een tekstverwerker om een bericht toe te voegen in
gedit "$pad/tempMail.txt"

read -p 'geef een onderwerp voor de mail: ' onderwerp
echo "De mail wordt vestuurd naar: $email."
mutt -s $onderwerp $email -a "$pad/KaripidisChari_$naam$voornaam.zip" < "$pad/tempMail.txt"
echo "De mail is succesvol verzonden naar $email."
```

```
45
46 rm "$pad/zip.txt" "$pad/KaripidisChari_$naam$voornaam.zip" "$pad/tempMail.txt"
47
48 read -p 'Wenst u het backup bestand terug te plaatsen?' backup
49
50 if [[ "$backup" == "J" ] -o [ "$backup" == "j" ]]
51         then mv $pad/GPGbackup.tex $pad/GPG.tex
52 fi
53
54
```