HONEY-POT MAIL ALERTS

Created by Daniel Tredler
GitHub link: https://github.com/TDanny/CS_Bsc_Technion-
LinkedIn Profile: https://www.linkedin.com/in/daniel-tredler-06761b213/

MAY 25, 2023

About the project

This project came to life because of my exposure to cyber security risks. In this project I modified existing and running platform called T-pot, a system that runs number of different honey-pots and collects data from each one. T-pot managing and overseeing.
T-pot collects all data about cyber security risks that each honey-pot detect, by using specific signs to detect which attack /actions were preformed against our network. T-pot is more learning tool then alerting tool , and that's why I saw potential .
I choose to modify T-pot by creating real time mail alerts when attacker trying to preform **bruteforce/dictionary attack** on T-pot's sign-in page.
There are many ways to create this functionality ,it took me 4-5 different attempts , each one I read about across the internet, but something was missing each time.
I decided to create my own scripts and to use a few method which I learned along the way.

Finally, I created this manual to help you and your organization to have a detection layer which can alert and notify about potential risk ahead. It is crucial to detect malicious behavior as soon as possible to prevent penetration.
You can use my scripts which I wrote at the manual and modify them as you wish, this is the base.

You can connect me using LinkedIn profile at the cover of this project for any questions.
Thanks for reading.
Let's get started!

Step 1 - Install any Honey-Pot .

I used T-pot system which includes few honeypots and it runs on Debian 11 Bullseye version.
Please, if you choose to work with T-pot read the README to understand the structure.
T-pot is a learning tool, but I modify it to be also alerting tool using proper Linux/Debian services and no other external tools(like cali /elastic/Kibana – which is possible as well).
If you decide to work on other system / unique honeypot follow the steps and It may work as well.

Important: **All my experience was on Debian, I didn't try it on other OS - YOU CAN TRY.

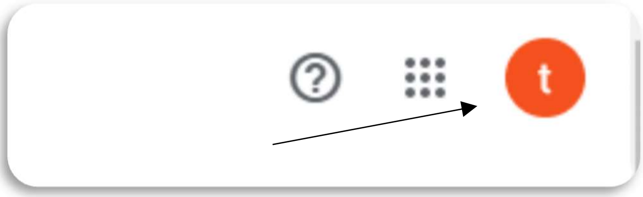
Link for T-pot (manual and files):
https://github.com/telekom-security/tpotce

Step 2 - Allow a third-party to enter our Email account and send emails.

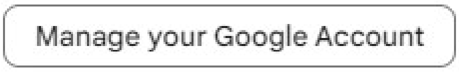
For that to happen we need to take a few steps , follow:

A. Enter to your mail account(I used Gmail and its working).

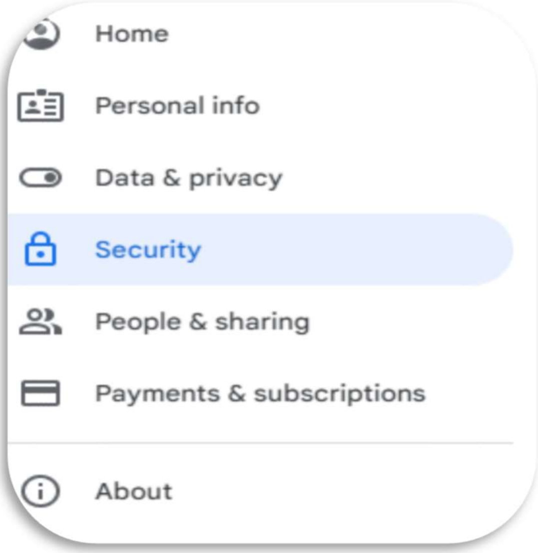
B. Click on :



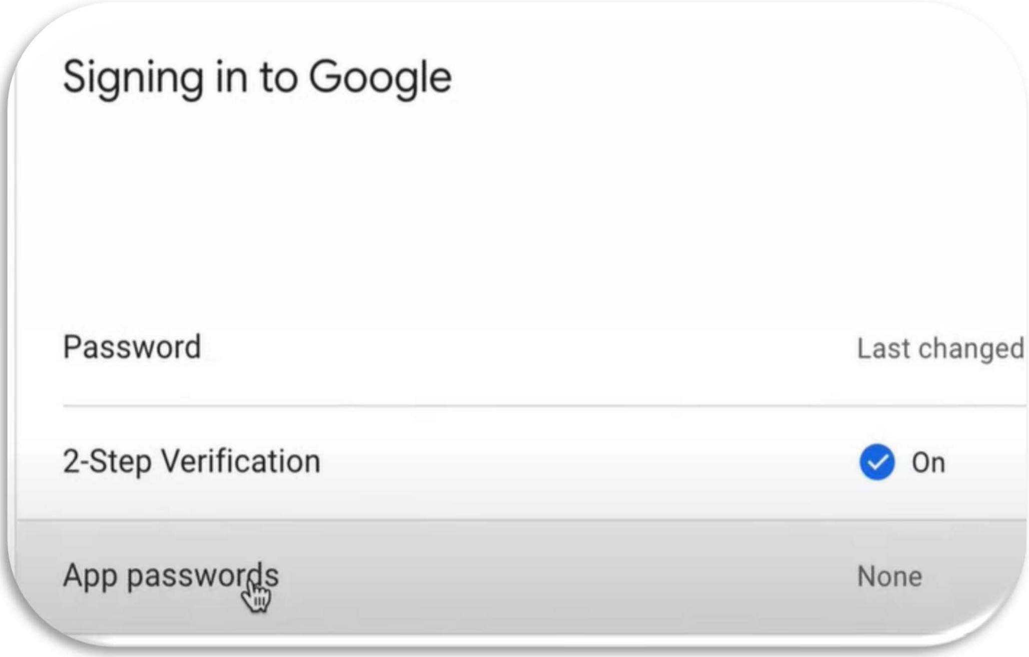
C. Then click on :



D. At the left menu , Click on "Security"



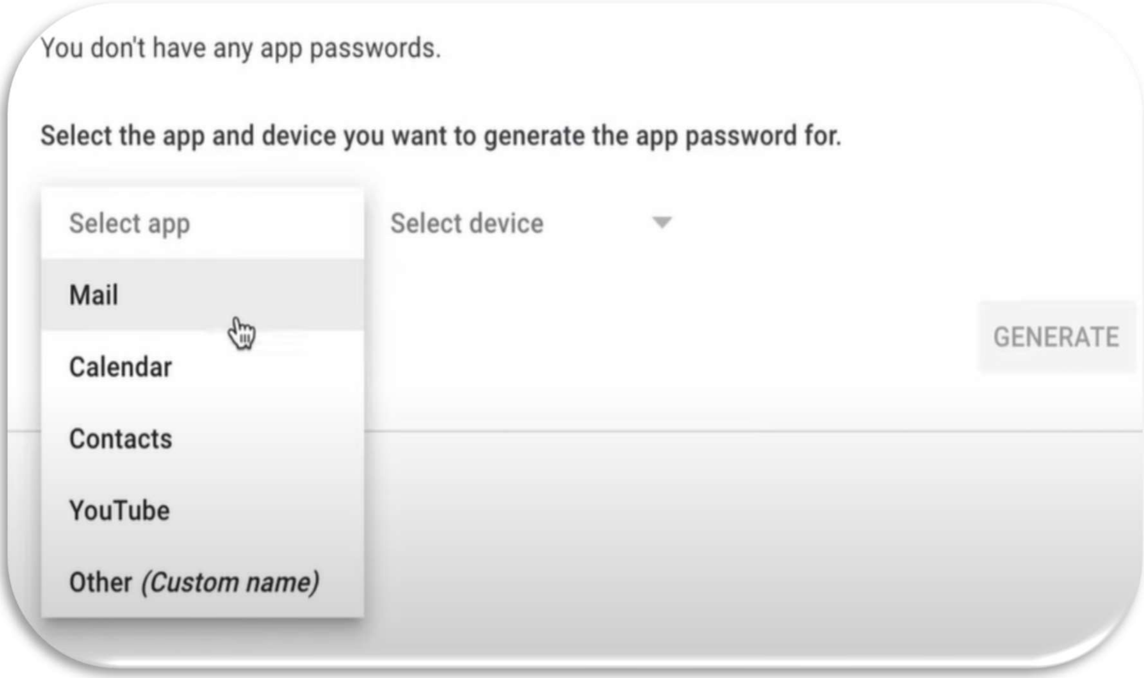
E. Create a 2-Step Verification by clicking "Turn on" , YOU MUST DO IT if you want to create a app password.
F. At the same "Security" page , scroll down and search for "App passwords" , click on it.
If you don't find it just search "App passwords" at the top search bar.



Signing in to Google

Password	Last changed
2-Step Verification	✓ On
App passwords	None

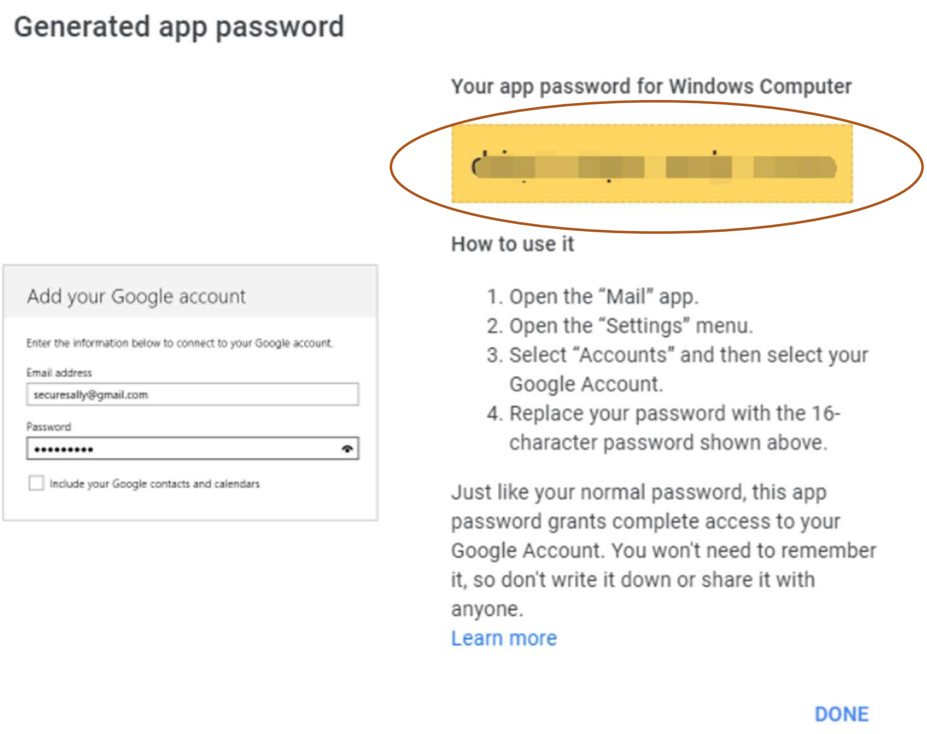
G. Select the type of App you want to connect. You can choose whatever you wish for ,its not that important, its mainly used to organization and order.



You don't have any app passwords.

Select the app and device you want to generate the app password for.

H. Click on "GENERATE" , that password will provide the third-party app access to this Email account.
write it down and click on "DONE".



Generated app password

Your app password for Windows Computer

[redacted]

How to use it

1. Open the “Mail” app.
2. Open the “Settings” menu.
3. Select “Accounts” and then select your Google Account.
4. Replace your password with the 16-character password shown above.

Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone.
Learn more

Add your Google account

Enter the information below to connect to your Google account.

Email address
securesally@gmail.com

Password
•••••••••

□ Include your Google contacts and calendars

Step 3 - install Postfix Service on our OS and Configure it:

A. Update and Upgrade your OS libraries:

```
sudo apt-get update && sudo apt-get upgrade
```

B. Install Postfix:

```
sudo apt-get install postfix
```

C. Configure SASL with the Gmail Account :
on your Honeypot we will create a new file at :

```
/etc/postfix/sasl/sasl_passwd
```

then, edit it by "nano" or "vi" and insert:

```
[smtp.gmail.com]:587 example@gmail.com:password
```

save and exit.
important: the password should be the one we generated in step 2.

D. Create a hash database file by using the next command , it will create a new file "sasl_passwd.db":

```
sudo postmap /etc/postfix/sasl/sasl_passwd
```

E. We need to set protective measures on our new passwd.db and therefore we need to use the next two commands:

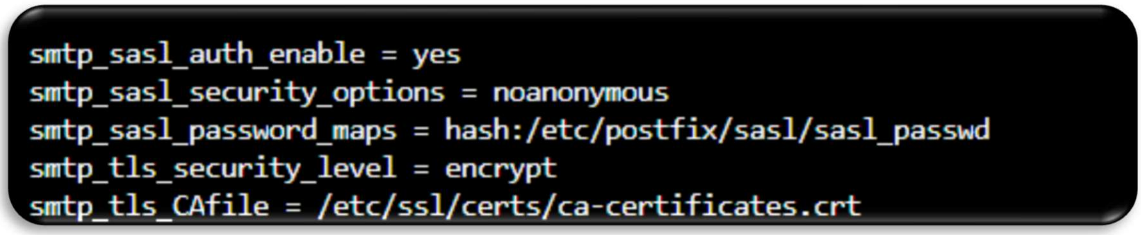
```
sudo chown root:root /etc/postfix/sasl/sasl_passwd /etc/postfix/sasl/sasl_passwd.db

sudo chmod 0600 /etc/postfix/sasl/sasl_passwd /etc/postfix/sasl/sasl_passwd.db
```

F. "nano" Or "vi" the next file : /etc/postfix/main.cf , lines: 41-43 , set the next values :

```
mydestination = $myhostname, localhost, localhost.localdomain
relayhost = [smtp.gmail.com]:587
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
```

G. Add to the end of that file (/etc/postfix/main.cf) the next lines , which will enable SASL authentication with Postfix service:



```
smtp_sasl_auth_enable = yes
smtp_sasl_security_options = noanonymous
smtp_sasl_password_maps = hash:/etc/postfix/sasl/sasl_passwd
smtp_tls_security_level = encrypt
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
```

H. Restart Postfix service by using the next command:

```
sudo systemctl restart postfix
```

I. You can verify that you installed and configure Postfix correctly by sendmail, use the next commands in the terminal:

```
sendmail example@gmail.com
To: example@gmail.com
Subject: Test Postfix service
Testing the new service!!
^d
```

Important notes:

** If terminal messaging an a error like :”sendmail command is not exist”,
try (full path): /usr/sbin/sendmail example@gmail.com
** **(^d)** is "ctrl+d" , which will notify the sendmail command that this is the EOF (end of file).
** after each line click ENTER.
** In this example I'm sending the mail from and to the same mail.
** The forth line is the message .

Step 4 – Create two Bash scripts that will check the Logs files:

First script : create a new shell file by using the next command:

```
nano extractLogs.sh
```

then write the next script:

```
grep “authentication failure;” /var/log/auth.log > /home/<username>/allFailure
grep “cockpit” /home/<username>/allFailure | tail -1 > /home/<username>/currentFailure
```

Important: **<username> is the host you are logged with (no root), so don’t write <username>, write the real name of the HOST.

example: `grep “cockpit” /home/`**daniel**`/allFailure | tail -1 >`

As you can see and realize , the two command in the first script will extract all failed logs and then extract the latest failed login attempt.

Second Script: create a new shell file by using the next command:

```
nano mailAlert.sh
```

then write:

```
MSG_CURRENT=$(grep “” /home/<username>/currentFailure)

DIFF=$(diff -w -B /home/<username/lastFailure
/home/<username>/currentFailure)

cmp /home/<username>/lastFailure /home/<username>/currentFailure >
/home/<username>/cmpFile
DIFFER=$(grep “differ” /home/<username>/cmpFile)
if [ “$DIFFER” != “” ]
then
cp /home/<username>/currentFailure /home/<username>/lastFailure
/usr/sbin/sendmail example@gmail.com <<EOF
To: example@gmail.com
Subject: Alert - Failed Login Attempt!!
Alert!!
Log message: “$MSG_CURRENT” .
DIFFERENCE” = $DIFFER” .
EOF
fi
cmp /home/<username>/WrongUserName /home/<username>/LastWrongUserName >
/home/tsec/cmpUsers
DIFFERENCE=$(grep "differ" /home/<username>/cmpUsers)
FAILED_USERS=$(cat /home/<username>/WrongUserName)
if [ "$DIFFERENCE" != "" ]
then
cp /home/<username>/WrongUserName /home/<username>/LastWrongUserName
/usr/sbin/sendmail tpotalerts@gmail.com <<EOF
To: tpotalerts@gmail.com
Subject: Failed Login Attempts - Users
User names which tried and failed to login T-pot system:
"$FAILED_USERS"
EOF
fi
```

Explanation about the mailAlert.sh script:
We will save the latest failed login attempt and we will compare it will the current failed login attempt. If there is any difference between the two then I want to get notify by mail. Else, don't do anything - the latest failed login is the same as the current fail – means it is the same failure attempt which we got a mail alert already .
The message contains IP and MAC address of the computer which carried those actions.
Important: copy paste the scripts, if you miss/add a blank space it may not work.

Step 5 – make the scripts executables by using the next commands:

```
sudo chmod +x /home/<username>/extractLogs.sh
sudo chmod +x /home/<username>/mailAlert.sh
```

Step 6 - create necessary files for the scripts, by "touch" command:

```
cd /home/<username>/
touch lastFailure
touch currentFailure
touch cmpFile
touch allFailure
touch LastWrongUserName
touch WrongUserName
touch cmpUsers
```

Step 7 – use CRON service

CRONTAB is a tool that helps to create a new cron jobs for CRON service.
That allows you to schedule and execute scripts in any given time.
I recommend to read about it.

Please , be aware that each HOST has its own CRONTAB.
Therefore:
The first Script "extractLogs.sh" needs to run by root because it access to logs files of the system which require higher permissions.

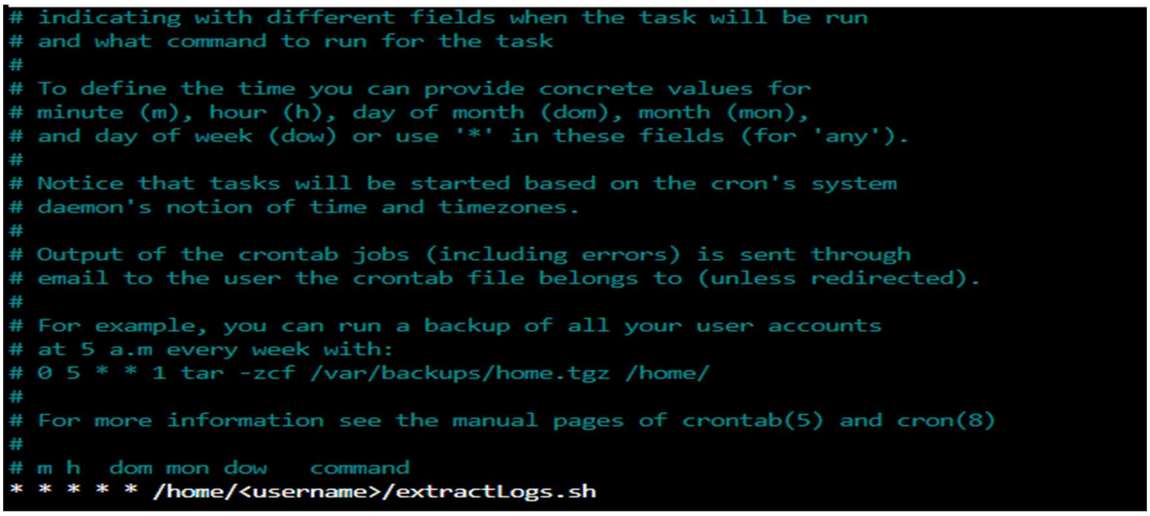
The second Script "mailAlert.sh" will run by <username> Host – the one each is logged.
How to do it? In the terminal write the next Command:

```
sudo crontab -e
```

It will open root's crontab file which you will need to modify, set the time you want the script to run and script's path.
Write it at the end of the file:

```
* * * * * /home/<username>/extractLogs.sh
```

(I attached a screenshot at the next page)



```
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
* * * * * /home/<username>/extractLogs.sh
```

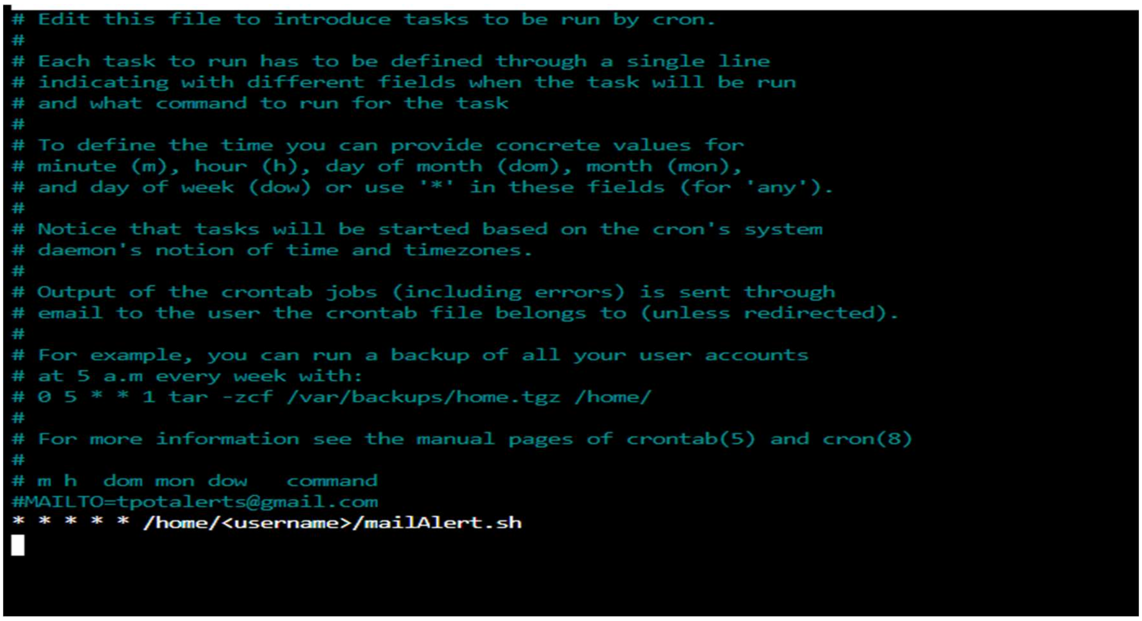
* * * * * means schedule a new cron job and execute it every minute. If you want to execute it on different time just read about crontab on google and set it correctly.

Now , we will do the same but for crontab of the host (<username> host) which will run the second script every minute: (same command but without “sudo” , it will open current HOST cron file)

```
crontab -e
```

It will open the host’s cron file: (-e stands for “edit”), then write at the end of the file :

```
* * * * * /home/<username>/mailAlert.sh
```



```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
#MAILTO=tpotalerts@gmail.com
* * * * * /home/<username>/mailAlert.sh
```

Step 8 – Restart crontab service

Use the next command to restart the service:

```
sudo service cron restart
```

OR

```
sudo service cron restart
```

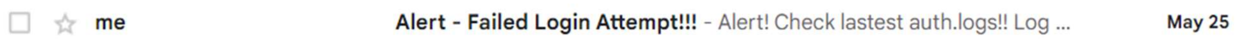
Please remember !!

Each time you see “<username>” in this manual it means you need to replace it with **your** HOST name, example:

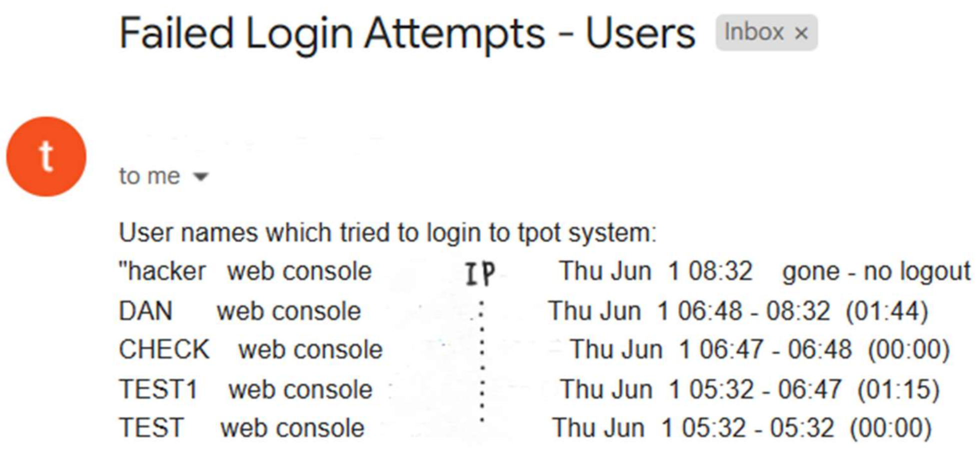
From: /home/<username>/mailAlert.sh

To: /home/Daniel/mailAlert.sh

Result:



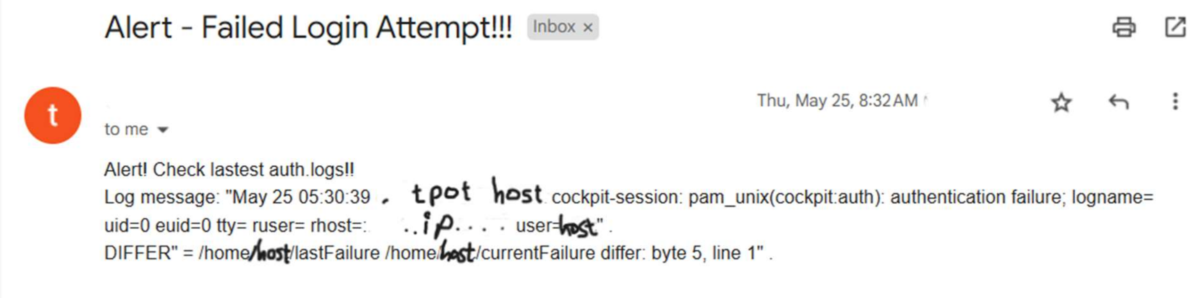
□ ☆ me **Alert - Failed Login Attempt!!!** - Alert! Check lastest auth.logs!! Log ... **May 25**



Failed Login Attempts - Users Inbox ×

to me

```
User names which tried to login to tpot system:
"hacker   web console      IP    Thu Jun  1 08:32   gone - no logout
DAN     web console        :    Thu Jun  1 06:48 - 08:32  (01:44)
CHECK    web console       :     Thu Jun  1 06:47 - 06:48  (00:00)
TEST1    web console       :    Thu Jun  1 05:32 - 06:47  (01:15)
TEST    web console        .    Thu Jun  1 05:32 - 05:32  (00:00)
```



Alert - Failed Login Attempt!!! Inbox ×

to me Thu, May 25, 8:32 AM

```
Alert! Check lastest auth.logs!!
Log message: "May 25 05:30:39 , tpot host cockpit-session: pam_unix(cockpit:auth): authentication failure; logname=
uid=0 euid=0 tty= ruser= rhost=: ..ip.... user=host" .
DIFFER" = /home/host/lastFailure /home/host/currentFailure differ: byte 5, line 1" .
```

As you can see , there are 2 types of mails you can receive.

One contains all usernames that the attacker used to login with, and the second is more specific and contains info about User name that exist but authentication failed due to wrong password.

- EOF -