BOX MAKING

process:

installer : continue without updating

keyboard : default

Type of installation: Ubuntu server

let the server use DHCP to self assign IP address to the machine

we will use entire disk

done

continue

Darkist

darkist

darkist

darkist1
darkist 1

Yes we will install openssh server

we will skip the rest

Twitter Account

Ghyophoggua

maloyik270@bitvoo.com

Ghyophoggua1337

https://twitter.com/ghyophoggua1

https://medium.com/@lihoxo4588/ghyophoggua-devs-are-fool-d130ae1f5f70

RSA ENCRYPTED PASSWORD:
P=103
Q=173

N = p*q : 17819

r = (p-1)*(q-1): 17544

Find a number equal to 1 mod r which can be factored
K : 526321
29*18149

e:18149
d: 29

```
    e   = 18149
    d   = 29
    N   = 17819
    r   = 17544
    e*d = 526321
e*d mod r = 1
e and r are relatively prime
d and r are relatively prime
```

```
MSG:3636
encrypted: 17446
Decrypted: 3636
```

Script

```
sudo su
kali
```

```
apt install apache2 -y
```

```
cd /var/www/html
```

```
apt install net-tools
```

```
apt install open vm tools
```

```
rm -rf /var/www/html/index.html
```

```
cd /var/home/www/
```

using script:

```
mkfolder()
{
names=(Ghyophoggua1 Ghyophoggua2 Ghyophoggua3 Ghyophoggua4 Ghyophoggua5
Ghyophoggua6 Ghyophoggua7 Ghyophoggua8)

# Loop through the array and create a folder for each name
for name in "${names[@]}"; do
  mkdir "$name"
done
}
cd  Ghyophoggua1

mkfolder

cd ../Ghyophoggua2

mkfolder

cd ../Ghyophoggua3
mkfolder

cd ../Ghyophoggua4

mkfolder

cd ../Ghyophoggua5
mkfolder


cd ../Ghyophoggua6

mkfolder

cd ../Ghyophoggua7

mkfolder


cd ../Ghyophoggua8

mkfolder


cd ../Ghyophoggua7

cd Ghyophoggua7




echo “ Hey! matt you know how only you know aboout this folder!   ya i have secretly kept
rest  of the password in this place        don't worry since you know we have heard RSA is
unbreakable by quantum computers and you have the private exponent , so NO worry
BTW  i have made the cipher  17446   with  e : 18149 and N :  17819   to be super safe
BYE!”  > ThisIsForMatt.txt
```

```
mkdir /var/ftp

apt install ufw
ufw status
ufw enable
ufw status
```

```
apt install vsftpd
cp /etc/vsftpd.conf /etc/vsftpd.conf.orig
nano /etc/vsftpd.conf
```

```
echo “# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=YES
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
#listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES
no_anon_password=YES
local_root=/var/ftp
# Uncomment this to allow local users to log in.
#local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
#anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
#anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES
#
# If enabled, vsftpd will display directory listings with the time
# in  your  local  time  zone.  The default is to display GMT. The
# times returned by the MDTM FTP command are also affected by this
# option.
use_localtime=YES
#
# Activate logging of uploads/downloads.
xferlog_enable=YES
#
# Make sure PORT transfer connections originate from port 20 (ftp-data).
connect_from_port_20=YES
#
# If you want, you can arrange for uploaded anonymous files to be owned by
# a different user. Note! Using "root" for uploaded files is not
# recommended!
#chown_uploads=YES
#chown_username=whoever
#
# You may override where the log file goes if you like. The default is shown
# below.
#xferlog_file=/var/log/vsftpd.log
#
# If you want, you can have your log file in standard ftpd xferlog format.
# Note that the default log file location is /var/log/xferlog in this case.
#xferlog_std_format=YES
#
# You may change the default value for timing out an idle session.
#idle_session_timeout=600
#
# You may change the default value for timing out a data connection.
#data_connection_timeout=120
#
# It is recommended that you define on your system a unique user which the
# ftp server can use as a totally isolated and unprivileged user.
```

```
#nopriv_user=ftpsecure
#
# Enable this and the server will recognise asynchronous ABOR requests. Not
# recommended for security (the code is non-trivial). Not enabling it,
# however, may confuse older FTP clients.
#async_abor_enable=YES
#
# By default the server will pretend to allow ASCII mode but in fact ignore
# the request. Turn on the below options to have the server actually do ASCII
# mangling on files when in ASCII mode.
# Beware that on some FTP servers, ASCII support allows a denial of service
# attack (DoS) via the command "SIZE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
#ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
# You may restrict local users to their home directories.  See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
#chroot_local_user=YES
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
#chroot_local_user=YES
#chroot_list_enable=YES
# (default follows)
#chroot_list_file=/etc/vsftpd.chroot_list
#
# You may activate the "-R" option to the builtin ls. This is disabled by
# default to avoid remote users being able to cause excessive I/O on large
# sites. However, some broken FTP clients such as "ncftp" and "mirror" assume
# the presence of the "-R" option, so there is a strong case for enabling it.
#ls_recurse_enable=YES
#
# Customization
```

```
#
# Some of vsftpd's settings don't fit the filesystem layout by
# default.
#
# This option should be the name of a directory which is empty.  Also, the
# directory should not be writable by the ftp user. This directory is used
# as a secure chroot() jail at times vsftpd does not require filesystem
# access.
#secure_chroot_dir=/var/run/vsftpd/empty
#
# This string is the name of the PAM service vsftpd will use.
pam_service_name=vsftpd
#
# This option specifies the location of the RSA certificate to use for SSL
# encrypted connections.
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
ssl_enable=NO
#
# Uncomment this to indicate that vsftpd use a utf8 filesystem.
#utf8_filesystem=YES”    >>  /etc/vsftpd.conf
```

```
ufw allow 20/tcp
ufw allow 21/tcp
ufw status
ufw allow ftp-data

cp ../file/AbsolutelyEmptyAndUselessAudio.wav   /var/ftp
```

```
echo “

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/
bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile  .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
```

```
# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
KbdInteractiveAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no

# GSSAPI options
#GSSAPIAuthentication no
#GSSAPICleanupCredentials yes
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the KbdInteractiveAuthentication and
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via KbdInteractiveAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and KbdInteractiveAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes
#TCPKeepAlive yes
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none
```

```
# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

# override default of no subsystems
Subsystem sftp          /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server
PasswordAuthentication yes”  >>   /etc/ssh/sshd_config
```

```
/sbin/service sshd status
```

```
/sbin/service sshd start
```

```
ufw allow 22/tcp
```

```
sudo useradd Ghyophoggua
sudo passwd  Ghyophoggua
echo “Super1337P@$$" -n
echo “Super1337P@$$" -n
```

```
mkdir /home/Ghyophoggua
```

```
chmod +777 /var/log/apache2/access.log


cd /var/www/html/


mkdir  zikimokbaka
cd  zikimokbaka
mkdir yakayaz

echo  “<?php  // Get the input from the URL parameter "input" $input =
$_GET['input'];  // Open the file "access.log" for writing $file = fopen(
"access.log", "a");  // Write the input to the file fwrite($file, $input . "\n"
);  // Close the file fclose($file);  ?>”              >> access.php

touch access.log
chmod +777 access.log

sudo apt-get autoclean

sudo apt-get clean

sudo add-apt-repository ppa:ondrej/php


sudo apt update

apt install php7.3 -y

sudo apt install libapache2-mod-php7.3

sudo a2enmod php7.3


sudo apt-get purge apache2

sudo apt-get purge apache2


// will upload and copy the python file to home/ghy***
```

```
cd /home/Ghyophoggua
chmod u+s  dark.py


echo “ghyophoggua ALL=(root) NOPASSWD: /usr/bin/python3* /home/Ghyophoggua/dark.py” >> /etc/sudoers

echo 'Will_you_Hecker' | md5sum > /home/Ghyophoggua/local.txt


echo 'you_Hecker' | md5sum > /home/darkist/proof.txt
```

walkthrough

```
        #!/bin/bash

# Open a new bash shell with root privileges
sudo bash

# Do not drop privileges
 bash m.sh


 echo -e "Super1337P@\$\$\nSuper1337P@\$\$" | sudo passwd Ghyophoggua
```

planning

website with many directory non-brute forceable

one of the directories will contain a file with password for the for user1(weird name on twitter) encrypted in rsa algorithm , now both ftp and ssh will be open

the username and password will access ftp which will only have read permission

now the ftp will contain an image which on steganography will give another password

which will be the password for ssh

once player enter ssh , they will have to find a way to get to the other user2 , whoes passsword will be seen in background of a image in victims twitter account
use wll have permission to run a particular file but would be no use for changing users

once the player switches to user2 we can see that there will be a bash script read only which will be taking command from a particular file base64 decoding it and the running checks against the particular
file name list // user will have to manipulate the list in such a way that it number of lines and words remain same but also priviledge escalates to root , finally user will get the root flag

files