Mission Name

ReconCar

History Context

Claire and Ethan are inside a cave trying to search the Recon CAR for the bug left by Dr. Pinche.

Technical High-Level Overview

A Smart Fridge forensic image is provided to the player. The goal of this challenge is to identify which bug was left by Dr.Pinche inside the Reconcar. In this scenario player will be investigating filesystem to locate any clue to a backdoor. Finally, considering filesystem dates and forensic artifacts related to persistence, the challenge could be solved.

Short Description

You´re going to analyse ReconCar system. This time ReconCar has a Raspberry Pi connected to infotainment system. Your goal will be to locate any identifier which involves Dr.Pinche´s malicious activity.

Mission Description

The goal of this challenge is to identify which bug was left by Dr.Pinche inside the ReconcPr. In this scenario you will be investigating a filesystem to locate any clue to a backdoor or other method which involves Dr.Pinche´s malicious activity. Your goal will be to locate the identifier, which leads that Dr.Pinche tampered ReconCar.

Location

RECON CAR - AIR / ALTAI MOUNTAINS

Tools

- Autopsy
- Openssl

Questions

Which program was used to modify the profile?

- nano

What was the last command launched by the user pi related to store passwords?

- lmpas gui

Items

1. Use Autopsy to mount evidence.
2. Identify all persistence Linux artifacts.
3. Use comments as passwords.

Write Up

First of all, player will have to mount the evidence provided using Autopsy:

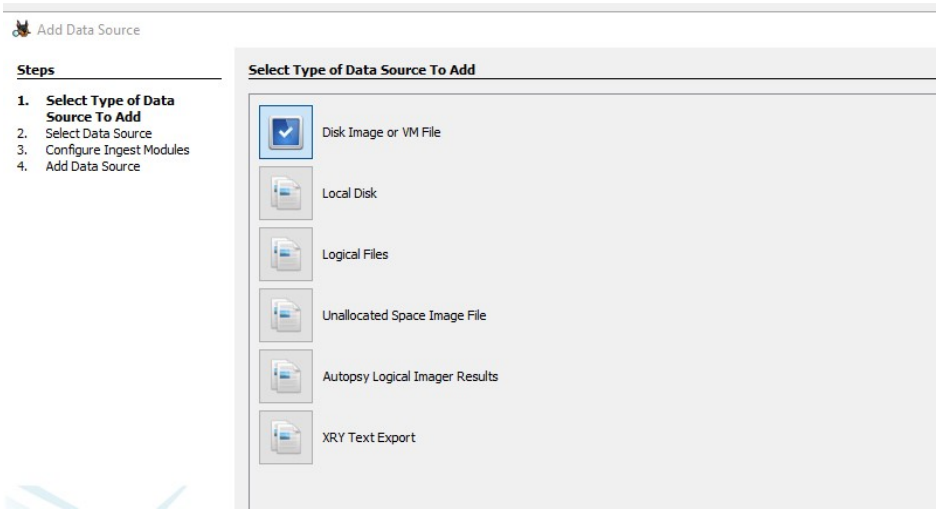


Figure 1

In this scenario, there is no need to index the evidence, just explore it. So, unselect all ingest modules,

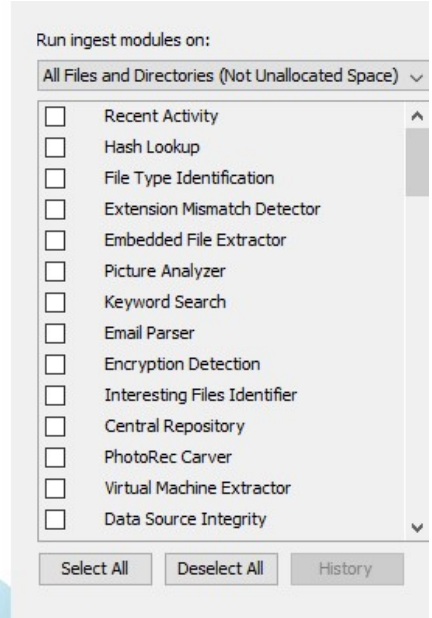


Figure 2

Then, player must select filesystem, in order to check files and folders:

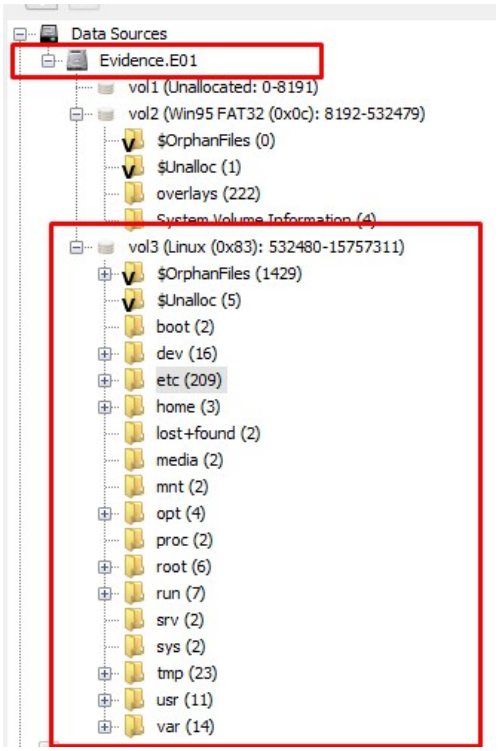


Figure 3

One possible wa, would be to analyse the bash_history file, to check commands launched, but as you can see below, this way could raise the complex of the case: there are two tools related to save secure passwords assword an impass.

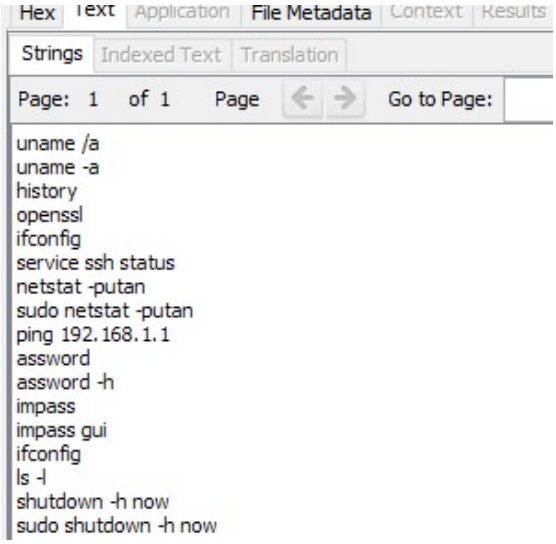


Figure 4

Considering the challenge description, Dr.Piche could hide something into ReconCard, so player could think about all persistence artifact in Linux. One place to check would be /etc/shadow to see, if there is any user related to Dr.Pinche.

Figure 5

Other place to check would be /etc/crontab

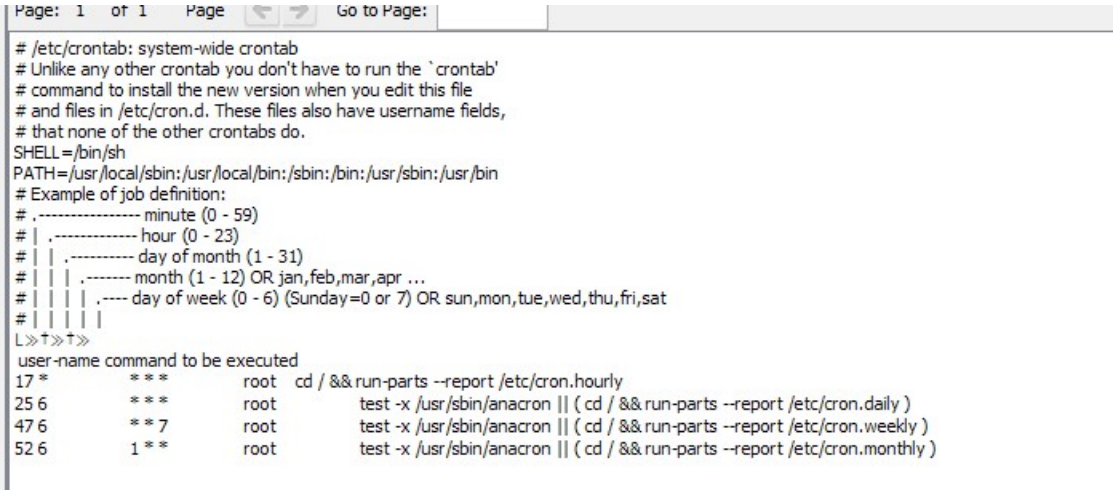


Figure 6

Another place would be /home/pi/.bash_rc:

```
    ,,
esac
# enable color support of ls and also add handy aliases
if [ -x /usr/bin/dircolors ]; then
    test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"
    alias ls='ls --color=auto'
    #alias dir='dir --color=auto'
    #alias vdir='vdir --color=auto'
    alias grep='grep --color=auto'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'
# some more ls aliases
#alias ll='ls -l'
#alias la='ls -A'
#alias l='ls -CF'
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.
if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
```

Figure 7

Finally we´ve found a clue in /home/pi/.profile

```
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.
# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022
# if running bash
if [ -n "$BASH_VERSION" ]; then
   # include .bashrc if it exists
   if [ -f "$HOME/.bashrc" ]; then
                . "$HOME/.bashrc"
   fi
# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ] ; then
   PATH="$HOME/bin:$PATH"
# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/.local/bin" ] ; then
   PATH="$HOME/.local/bin:$PATH"
cat treh.txt.enc | openssl aes-256-cbc -d -a); exit
#Always use the same:cjklmno
```

Figure 8

This commands inside the **.profile** file, could have been modified by Dr.Pinche:

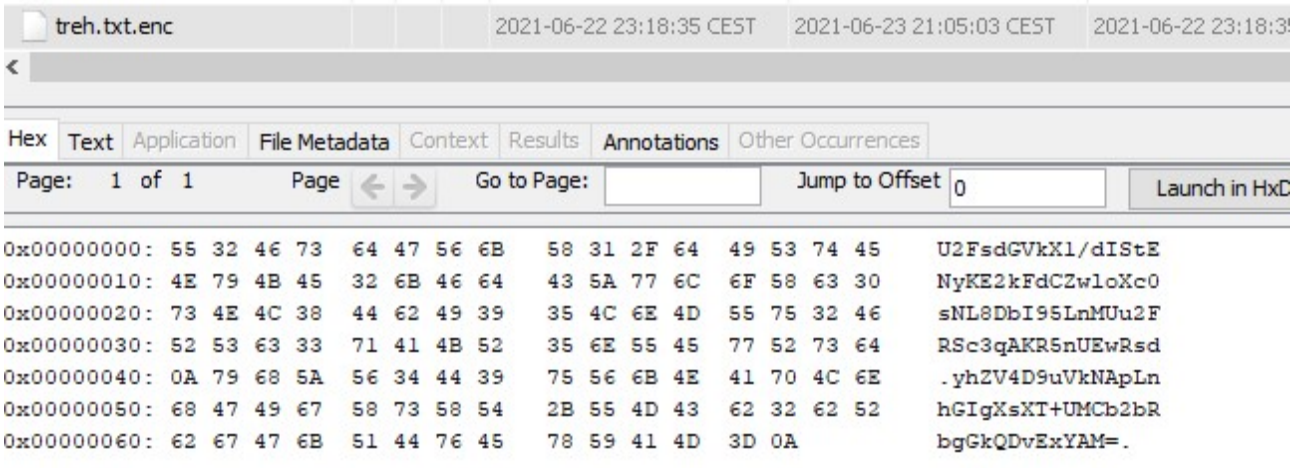


Figure 9

If we check file treh.txt.enc

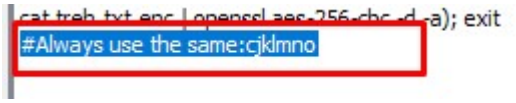
```
jmma@demowindows:/mnt/c/THREATIA/C3-M2/scripts$ file treh.txt.enc
treh.txt.enc: openssl enc'd data with salted password, base64 encoded
```

Figure 10

So, the file contains encrypted data. If we launch the command:

- cat treh.txt.enc | openssl aes-256-cbc -d -a

We will need the password to open the encrypted piece of information. So, the key is the comment on the profile file:



Finally we can open the container and get flag: 93203023208

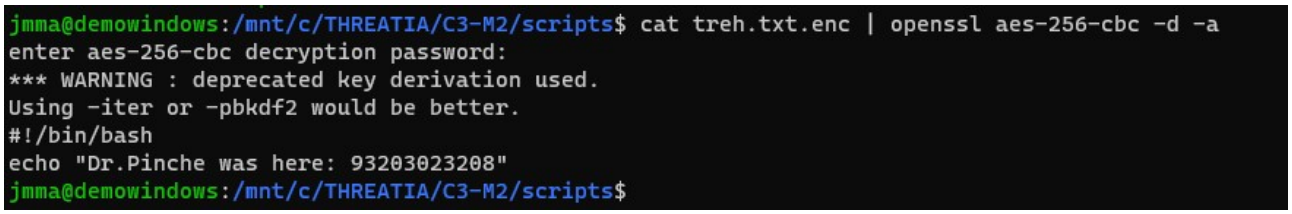


Figure 11

Flag Information

flag{93203023208}