Conversation with:

Nasreen Al Najjar
Systems Security Specialist

Systems Security Analysis Challenge Chat Transcript

Nasreen sent:
Welcome, @playerone. It looks like you're my new intern!

I'm Nasreen Al Najjar, and I work as a Systems Security Specialist here at TryCyber Managed Service Provider (MSP).

Nasreen sent:
I'll introduce you to some of the core **Systems Security Analysis** tasks. For your first day, I'm going to have you assist me with a few of my simpler tasks on the **Ubuntu 22.04 Linux** workstation you are looking at now.

Nasreen sent:
The tasks we will be doing today will be completed using the **Terminal** and a **text editor**. The Terminal can be intimidating if you have not used it before. So let me know if you want a brief tutorial on it.

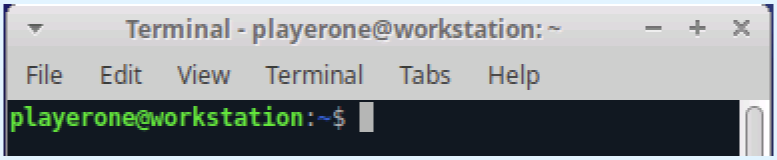
Nasreen sent:
Shall we get started? 😊

Participant sent:
Terminal tutorial, please!

Nasreen sent:
The **Terminal** is primarily used to access an application called a shell. We interact with shells via a **command-line interface (CLI)** where we input and execute text-based commands on a system.

Nasreen sent:
Let's open the Terminal on this system now and run a few basic commands so you get the hang of it. You can open the **Terminal** by double-clicking the icon on the desktop named **Terminal Emulator**. It should look like this once you have it open...

Nasreen sent:



```
playerone@workstation:~$ 
```

Participant sent:
I've opened the Terminal.

Nasreen sent:
Great. Basically, you type commands into the shell within the Terminal and then hit `Enter` or `Return` on the keyboard to run the command.

Nasreen sent:
To properly utilize any CLI command, you must understand its format, options, and other arguments. The **format** defines the structure and order of options and other arguments, the **options** define or modify behavior, and the **arguments** are command defined input types (e.g., files, directories, users, software packages, etc.).

Nasreen sent:
Additionally, all CLI commands on this system are case-sensitive, so pay close attention to the capitalization of letters.

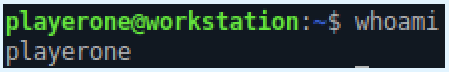
Nasreen sent:
The basic command format is...

```
command [OPTIONS] [ARGUMENTS]
```

However, it varies a lot from command to command, and options and other arguments are not always required.

Nasreen sent:
An example of a command that does not require any options or other arguments is `whoami`. If you type `whoami` into the shell and then hit `Enter` or `Return` on the keyboard, the `whoami` command will output the name of the user running the command into the Terminal. It should look like this if you run that command...

Nasreen sent:



```
playerone@workstation:~$ whoami
playerone
```

Participant sent:
What's a more involved example?

Nasreen sent:
I'll give you a more complex example using the `ls` command.

Nasreen sent:
In this example, we will use the `ls` command to list detailed information about the entire contents of the `Templates` directory (i.e., folder) in your (playerone's) `home` directory. Our example `ls` command will use the following format...

```
ls [OPTIONS] [DIRECTORY]
```

Note that in this case, the command's argument requires the input to be a directory.

Nasreen sent:
The actual command we want to run is...

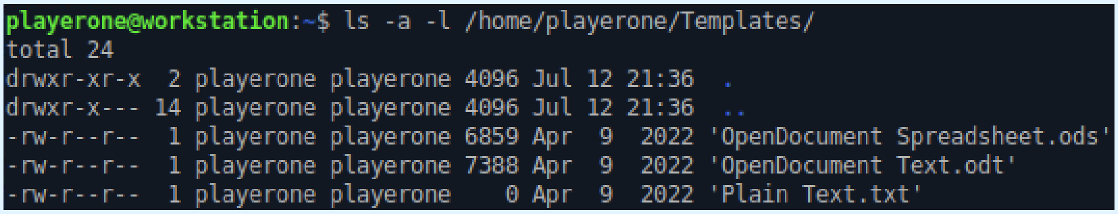
```
ls -a -l /home/playerone/Templates
```

Nasreen sent:
The `-a` and the `-l` are both options that modify the `ls` command's behavior, and `/home/playerone/Templates` is our argument which is a directory provided in the form of a **path**.

Nasreen sent:
For this command's argument, we must provide the **path** to the directory from the root of the file system; otherwise, the system won't know which directory named `Templates` we are referring to.

Nasreen sent:
And finally, when you run that command, the output should look like this...

Nasreen sent:



```
playerone@workstation:~$ ls -a -l /home/playerone/Templates/
total 24
drwxr-xr-x  2 playerone playerone 4096 Jul 12 21:36  .
drwxr-x--- 14 playerone playerone 4096 Jul 12 21:36  ..
-rw-r--r--  1 playerone playerone 6859 Apr  9  2022 'OpenDocument Spreadsheet.ods'
-rw-r--r--  1 playerone playerone 7388 Apr  9  2022 'OpenDocument Text.odt'
-rw-r--r--  1 playerone playerone    0 Apr  9  2022 'Plain Text.txt'
```

Participant sent:
Got it. Any extra notes?

Nasreen sent:
The Terminal and shells are incredibly powerful and versatile tools. Not all commands and programs follow the general structure I've provided you with here. Unfortunately, we only have time to cover the basics, but I do have a few last things I'd like to mention.

Nasreen sent:
Some commands will not print visual output to the Terminal in normal operation, such as the `cp` command, which is used to copy files and directories.

Nasreen sent:
Many commands can use multiple arguments, handle multiple input types for arguments (e.g., file and/or directory paths), or have options that will have their own arguments.

Nasreen sent:
Some options can, or must be, written in a long-form format (e.g., `ls --all [DIRECTORY]` is the same as `ls -a [DIRECTORY]`).

Nasreen sent:
Options without arguments can often be provided together and in any order (e.g., `ls -la [DIRECTORY]` is the same as `ls -a -l [DIRECTORY]`).

Nasreen sent:
And last, but certainly not least, you can almost always reference a command's format, options, and other arguments using the command `man [COMMAND]` (e.g., `man cp`) to view the provided command's manual page in the Terminal.

Nasreen sent:
Hopefully that was not too much information! I know it seems like a lot, but it gets easier the more you use it. For today's tasks, I'll be sure to provide you with more details for any commands and programs you'll need.

Participant sent:
Sounds good! I'm ready to get started!

Nasreen sent:
Perfect!

Nasreen sent:
Our job at TryCyber MSP is to deliver services, such as cybersecurity support and management, to our clients.

Nasreen sent:
The client we will be working for today had previously hired us to develop some baseline system security policies, a project I personally worked on. After they failed a recent audit, the same client has now hired us to implement those baseline system security policies since their staff is spread too thin to get it done.

Nasreen sent:
Critical to becoming effective at Systems Security Analysis is being deeply familiar with system security practices; these practices enable us to effectively review systems, write policies, and, in some cases, implement those policies. While we won't be writing any policies today, we will be implementing one that I wrote. 🙂

Nasreen sent:
So, let's discuss the purpose and design of these baseline system security policies before we implement one on this Ubuntu Linux workstation.

Nasreen sent:
The purpose of our baseline system security policies is to maintain or enhance a computer systems' confidentiality, integrity, and availability.

Nasreen sent:
As for their design, our policies include a name, technical and administrative reasoning, and technical instructions for checking and implementing the policy. The policy we will be looking at today has technical instructions in the form of embedded excerpts from matching **Vulnerability Identifiers (Vul IDs)** within applicable **Security Technical Implementation Guides (STIGs)**.

Nasreen sent:
STIGs, which are created by the Defense Information Systems Agency (DISA), are essentially guides that help you check systems for known vulnerabilities and fix those vulnerabilities when found.

Nasreen sent:
We will use those embedded STIG Vul excerpts today as policy implementation guides.

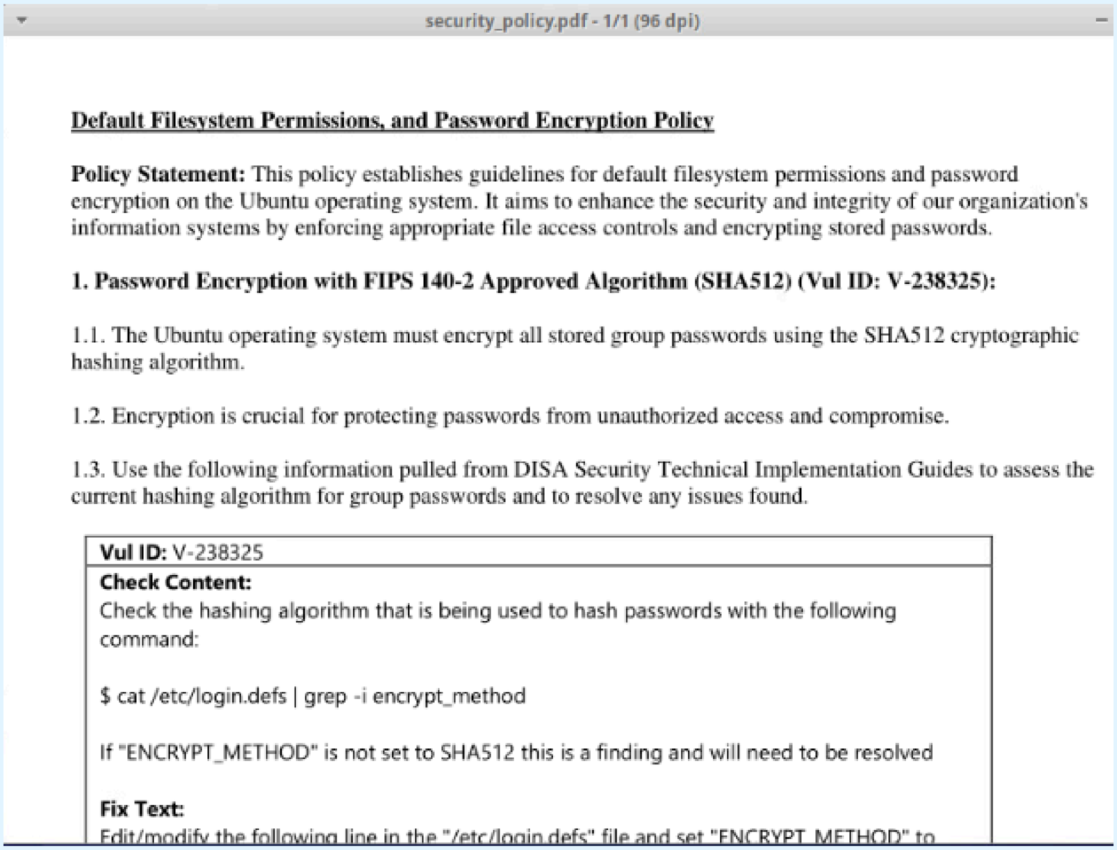
Participant sent:
Got it.

Nasreen sent:
Okay, with that little explainer out of the way, let's get to it!

Nasreen sent:
The policy document you will need to reference for today's tasks is in the `Materials` folder on the workstation desktop. It is a one-page PDF document named `security_policy.pdf`.

Nasreen sent:
To open this PDF document, you will want to double-click the `Materials` folder on the desktop, then double-click the `security_policy.pdf` file. This will open the PDF in the default PDF viewer on the workstation. It should look something like this once you have it open...

Nasreen sent:



Default Filesystem Permissions, and Password Encryption Policy

Policy Statement: This policy establishes guidelines for default filesystem permissions and password encryption on the Ubuntu operating system. It aims to enhance the security and integrity of our organization's information systems by enforcing appropriate file access controls and encrypting stored passwords.

1. Password Encryption with FIPS 140-2 Approved Algorithm (SHA512) (Vul ID: V-238325):

1.1. The Ubuntu operating system must encrypt all stored group passwords using the SHA512 cryptographic hashing algorithm.

1.2. Encryption is crucial for protecting passwords from unauthorized access and compromise.

1.3. Use the following information pulled from DISA Security Technical Implementation Guides to assess the current hashing algorithm for group passwords and to resolve any issues found.

Vul ID: V-238325

Check Content:
Check the hashing algorithm that is being used to hash passwords with the following command:

```
$ cat /etc/login.defs | grep -i encrypt_method
```

If "ENCRYPT_METHOD" is not set to SHA512 this is a finding and will need to be resolved

Fix Text:
Edit/modify the following line in the "/etc/login.defs" file and set "ENCRYPT_METHOD" to

Participant sent:
I've got it open!

Nasreen sent:
As you can see, this policy document has two sections, and while I'd encourage you to review the whole policy, we need to focus on the STIG Vul excerpts and the technical instructions within them to get this policy implemented.

Nasreen sent:
Each policy section contains a matching STIG Vul excerpt embedded at the bottom of the section within a box with a black outline. The STIG Vul excerpts start with the Vul ID (e.g., Vul ID: V-###), which is then followed by details on how to check for and fix the corresponding vulnerability.

Nasreen sent:
In today's case, fixing the vulnerability is also implementing a section of the policy. Consider each policy section a task I'd like you to help me complete today. 🙂

Nasreen sent:
So, with that, let's handle the first policy section listed in this document, **Password Encryption with FIPS 140-2 Approved Algorithm (SHA512) (Vul ID: V-238325)**. Let me know once you've read through it.

Participant sent:
I've reviewed it.

Nasreen sent:
Great!

Nasreen sent:
For this first policy section, we need to ensure the hashing algorithm used on this workstation to hash passwords is **set to SHA512**. To do that, we need to run the command provided in the **Check Content** section of the STIG Vul excerpt in the Terminal. The provided command being...

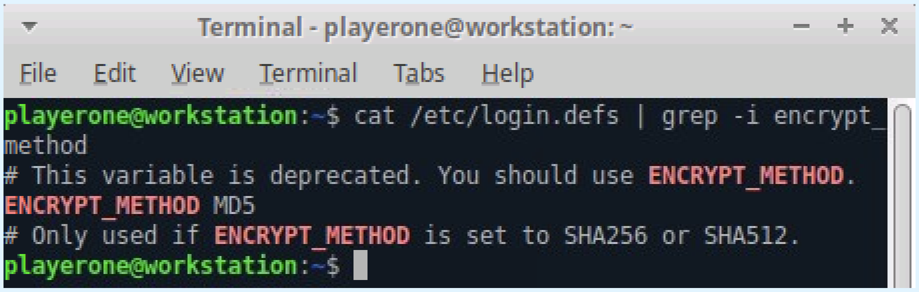
```
cat /etc/login.defs | grep -i encrypt_method
```

Note that the character between the end of the `cat` command and the beginning of the `grep` command is a pipe character (i.e., |) and not the letter l or the number 1. You can usually find the pipe character on your keyboard on the key above your `Enter` or `Return` key.

Nasreen sent:
The command the STIG Vul is having us run basically outputs all lines within the configuration file `login.defs` (which resides in the `etc` directory) that contain the case-insensitive pattern `encrypt_method`. While this is going to return some lines we don't care about (e.g., lines containing comments, which start with `#`), it will also have the line we do care about, which is formatted `ENCRYPT_METHOD [HASHING ALGORITHM]`.

Nasreen sent:
You should see it when you run the command, which has output that looks like this...

Nasreen sent:



```
playerone@workstation:~$ cat /etc/login.defs | grep -i encrypt_
method
# This variable is deprecated. You should use ENCRYPT_METHOD.
ENCRYPT_METHOD MD5
# Only used if ENCRYPT_METHOD is set to SHA256 or SHA512.
playerone@workstation:~$ 
```

Nasreen sent:
As expected, it looks like **ENCRYPT_METHOD** is not set to the hashing algorithm the policy calls for. Instead of using the SHA512 hashing algorithm, this workstation is currently using the MD5 hashing algorithm.

Participant sent:
What do we do now?

Nasreen sent:
Now its time to implement this policy! Lucky for us, we don't have to figure out how to implement it ourselves; the included **Fix Text** section of the STIG Vul excerpt tells us what we need to do.

Nasreen sent:
In this case, the Fix Text is directing us to **edit the `login.defs` configuration file** (located at `/etc/login.defs`) and update the line containing the **ENCRYPT_METHOD** configuration setting. To do that, you'll need to open the `login.defs` file with a text editor running with elevated permissions.

Nasreen sent:
There are many ways to do that, but I'll suggest just running the following command in the Terminal...

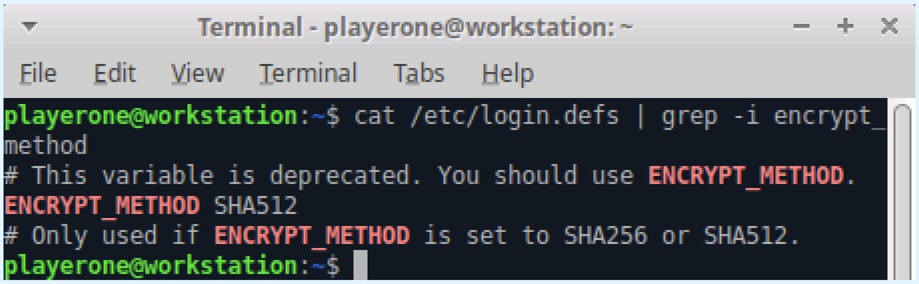
```
sudoedit /etc/login.defs
```

That command will open the file at the provided path, `/etc/login.defs`, with the default text editor (`mousepad` in this case) running as the superuser.

Nasreen sent:
When you use the `sudoedit` command, you will sometimes be prompted for your password. Just enter your password into the prompt and hit `Enter` or `Return` on the keyboard when you're done. (Note: Your password can be found on the Info Tab)

Nasreen sent:
Once you've **updated the file correctly and closed the text editor** (when using `sudoedit`, you must close the text editor and choose Save in the Save Changes dialog box for the file to be updated), you should be able to run the command in the Check Content section of the STIG Vul for this policy again and get this output...

Nasreen sent:


Terminal - playerone@workstation: ~
File Edit View Terminal Tabs Help

```
playerone@workstation:~$ cat /etc/login.defs | grep -i encrypt_method
# This variable is deprecated. You should use ENCRYPT_METHOD.
ENCRYPT_METHOD SHA512
# Only used if ENCRYPT_METHOD is set to SHA256 or SHA512.
playerone@workstation:~$
```

Nasreen sent:
Let me know when you are done or if you'd like some extra help. 😊

Participant sent:
Help, please!

Nasreen sent:
Of course!

Nasreen sent:
I will assume you have the `login.defs` file open in the `mousepad` text editor. If not, you will need to do so by running the `sudoedit` command I sent you earlier (i.e., `sudoedit /etc/login.defs`).

Nasreen sent:
The line in this configuration file that contains the ENCRYPT_METHOD configuration setting is line 284. You can scroll down until you find it, or you could use the Find feature in the `mousepad` text editor under the Search section of the menu bar and search for what the line contains right now (i.e., 'ENCRYPT_METHOD MD5').

Nasreen sent:
Once you find the right line in the configuration file, you need to update the line from reading...

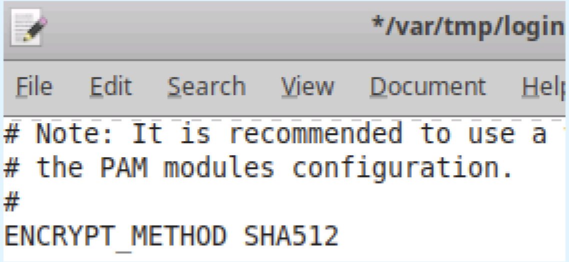
```
ENCRYPT_METHOD MD5
```

to read...

```
ENCRYPT_METHOD SHA512
```

Nasreen sent:
Basically, it should look like this...

Nasreen sent:



```
# Note: It is recommended to use a
# the PAM modules configuration.
#
ENCRYPT_METHOD SHA512
```

Nasreen sent:
Then you will need to save the file and close the text editor; if you didn't save it before trying to close it, it will ask you if you want to save the changes. As a note, when using `sudoedit`, you must save and close the text editor before the changes are written to the open file.

Nasreen sent:
Once you've finished the update and confirmed it using the command in the Check Content section of the STIG Vul for this policy, let me know so we can keep going.

Participant sent:
I'm done.

Nasreen sent:
Great Job! ⭐

Nasreen sent:
Now, I've got good news and bad news. The bad news is I must assist another client with an urgent issue and will be unreachable for the rest of the day. The good news is that all you have to do to finish today's tasks is repeat the policy implementation process for the final policy section of the `security_policy.pdf` document, the **Default Filesystem Permissions (Vul ID: V-238209)** section.

Nasreen sent:
I skimmed the policy section already, and I'm sure you can handle it. The **Check Content** section of the STIG Vul for the **Default Filesystem Permissions (Vul ID: V-238209)** section uses a simpler command than the previous policy, and the **implementing this policy section via the Fix Text** requires edits to the same file (i.e., `login.defs`) using the same method (i.e., the `sudoedit` command to open the file in the text editor) as the previous policy section.

Nasreen sent:
Alright, I've got to go. Thanks for all your help! Good luck!