

Lab 8 Report

Jason Tolbert

The Pennsylvania State University

IST 894-001: Capstone Experience

Dr. Michael Bartolacci, Instructor

March 23rd, 2024

Table of Contents

General Overview.....	3
Cryptography.....	3
Access Control	4
Technical Overview	5
Cryptography.....	5
AAA.....	5
References	7
Screenshots	8

General Overview

This lab introduces participants to two foundational cybersecurity concepts: cryptography and access control.

Cryptography

The first half of the lab explores basic cryptographic techniques. Participants begin by exploring Steganography. Steganography is the practice of concealing secret information within nonsecret media. The nonsecret media can be text, images, videos, or just about anything else. Images are the most common choice, due to their ubiquity on the internet and the diverse array of methods that can be used to hide information within them (Johnson & Jajodia, 1998; Morkel et al., n.d.). Lab participants use command-line tools to extract hidden message from image files and understand how data can be covertly embedded and later recovered.

The lab then switches topics to encryption algorithms. Participants are introduced to both symmetric algorithms (in which the same key is used for both encryption and decryption) and asymmetric algorithms (in which the encryption key is different from the decryption key) (Yassein et al., 2017). The lab demonstrates these algorithms through the use of OpenSSL, a popular open-source SSL/TLS implementation that also includes a robust suite of cryptography tools.

Access Control

The second half of the lab focuses on access control — or Authentication, Authorization, and Accounting (AAA) to be specific. Much of this half lab focuses on securing remote access via SSH. Participants compare key pair-based SSH authentication with password-based SSH authentication and learn how key pairs provide significantly stronger security. Participants also explore best practices for securing SSH passwords (e.g., using a password manager and never reusing the same password for multiple machines) that they can apply in situations where key pair authentication is not an option.

This part of the lab also touches on Unix file permissions. On Linux and other Unix-like systems, each file has three categories of permissions: 1) permissions that apply only to a given user, 2) permissions that apply to a group of users, and 3) permissions that apply to users who don't fall in either of the other categories. Within each category, there are three permissions that can be set: permission to read the file, permission to write to the file, and permission to execute the file. (Grampp & Morris, 1984) Lab participants are taught how these permissions work, how to modify them, and how the ways different users are able to interact with files can change based on the permissions that have been set.

Technical Overview

This lab introduces participants to two foundational cybersecurity concepts: cryptography and AAA.

Cryptography

Participants begin the cryptography portion of the lab with an introduction to steganography. Steghide is used to extract a secret message from a JPEG file, illustrating to participants how information steganographic techniques can be covertly hide and later retrieved without obviously altering the cover medium. Following that, participants use the OpenSSL cryptographic suite to practice encrypting and decrypting files. They're shown both symmetric algorithms, like DES, 3DES, and AES, and asymmetric algorithms, like RSA and ECC.

AAA

The AAA portion of these lab teaches participant several ways of controlling and securing access to Linux systems. Naturally, this means a substantial part of it is focused on the Unix permission system. Participants get hands-on experience using `chmod`, `chown`, and `ls` to configure and inspect permissions for files and directories. Once they've been introduced the basic `r-w-x` syntax, they learn how to convert it to octal notation. They also

switch user accounts throughout the lab to see firsthand how the permissions they've set have changed the way different users can interacting with the filesystem.

The lab then pivots to the subject of SSH — specifically, best practices when it comes to SSH credentials. They learn how key-pair authentication trades usability for stronger security, and how password-based authentication prioritizes usability at the expense of security. They also spend time working with the KeePass password manager and learn how to use it to securely store SSH passwords in the event that they must be used over key pairs.

References

- Grampp, F. T., & Morris, R. H. (1984). The UNIX system UNIX operating system security. *AT&T Bell Laboratories Technical Journal*, 63(8), 1649–1672. AT&T Bell Laboratories Technical Journal. <https://doi.org/10.1002/j.1538-7305.1984.tb00058.x>
- Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. *Computer*, 31(2), 26–34. Computer. <https://doi.org/10.1109/MC.1998.4655281>
- Morkel, T., Eloff, J. H. P., & Olivier, M. S. (n.d.). *An overview of image steganography*.
- Yassein, M. B., Aljawarneh, S., Qawasmeh, E., Mardini, W., & Khamayseh, Y. (2017). Comprehensive study of symmetric key and asymmetric key encryption algorithms. *2017 International Conference on Engineering and Technology (ICET)*, 1–7. <https://doi.org/10.1109/ICEngTechnol.2017.8308215>

Screenshots

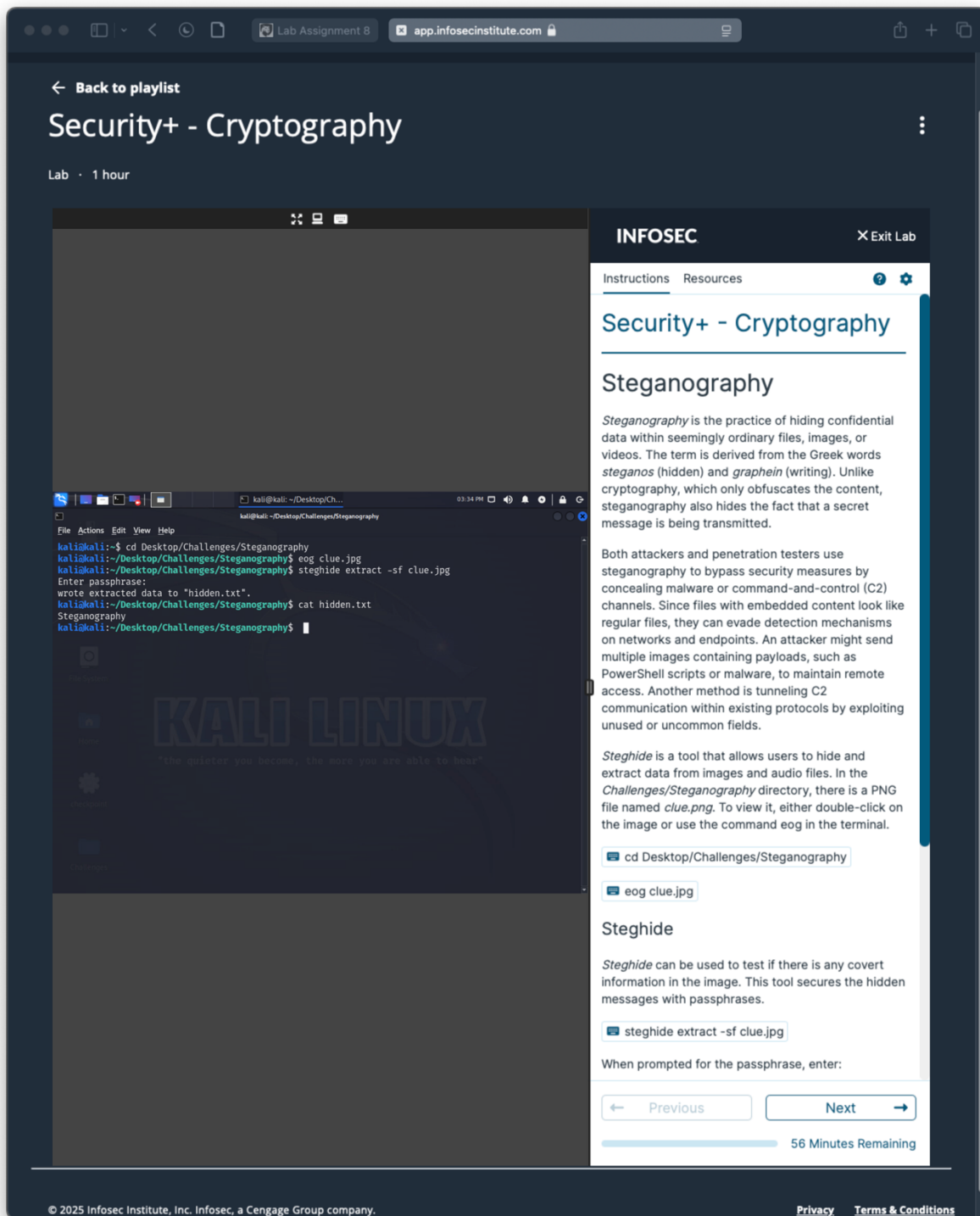


Figure 1. Viewing information hidden in the clue.jpg file.

Lab Assignment 8

app.infosecinstitute.com

Back to playlist

Security+ - Cryptography

Lab · 1 hour

INFOSEC

Exit Lab

Instructions Resources

comparison

In the terminal, navigate to *Cryptography* directory and list the file. Each of the files is the ciphertext of the word *Cryptography* encrypted using different symmetric algorithms and modes of operations. The command below displays the ciphertext in each file.

```
cd ../Cryptography/Symmetric
```

```
awk '{print}' * > salted.txt
```

The output of the command shows the difference in encryption.

```
cat salted.txt
```

OpenSSL

OpenSSL is a command-line utility that can be used to generate private keys, encrypt, decrypt, create CSRs, generate certificates, and identify certificate information. To decrypt the DES ECB ciphertext, use openssl.

```
openssl enc -d -des-ecb -in des-ecb.enc -out des-ecb.txt
```

When prompted for the password, type:

```
secure
```

To view the content of this file, use the following command:

```
cat des-ecb.txt
```

What message will be displayed after decrypting the ciphertext?

Check

Previous Next

55 Minutes Remaining

```
kali@kali: ~/Desktop/Challenges/Steganography
File Actions Edit View Help
kali@kali:~/Desktop/Challenges/Steganography$ eog clue.jpg
kali@kali:~/Desktop/Challenges/Steganography$ steghide extract -sf clue.jpg
Enter passphrase:
wrote extracted data to "hidden.txt".
kali@kali:~/Desktop/Challenges/Steganography$ cat hidden.txt
Steganography
kali@kali:~/Desktop/Challenges/Steganography$ cd ../Cryptography/Symmetric
kali@kali:~/Desktop/Challenges/Cryptography/Symmetric$ awk '{print}' * > salted.txt
kali@kali:~/Desktop/Challenges/Cryptography/Symmetric$ cat salted.txt
Salted_E.....ce0e2*1w
Salted_D.....ce0e2*1w
Salted_Re1e* "uJzeV4e9*cc
Salted_[***]e0V***"***[***]e
Salted_***e0e2e1e1eX***e1e
Salted_ *0 *1y1e***ij***
***

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric$ openssl enc -d -des-ecb -in des-ecb.enc -out
des-ecb.txt
enter DES-ECB decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
kali@kali:~/Desktop/Challenges/Cryptography/Symmetric$ cat des-ecb.txt
Cryptography
kali@kali:~/Desktop/Challenges/Cryptography/Symmetric$
```

© 2025 Infosec Institute, Inc. Infosec, a Cengage Group company.

[Privacy](#) [Terms & Conditions](#)

Figure 2. Decrypting DES-ECB ciphertext with OpenSSL.



Figure 3. Decrypting DES-CBC ciphertext with OpenSSL.

Back to playlist

Security+ - Cryptography

Lab · 1 hour

INFOSEC

Exit Lab

InstructionsResources

openssl enc -d -des-cbc -in des-cbc.enc -out des-cbc.txt

When prompted for the password, enter:

secure

Therefore, to view the content, use the following command:

cat des-cbc.txt

Decrypt 3DES ECB Cipher text

The command below decrypts the 3DES ECB ciphertext:

openssl enc -d -des-ede3-ecb -in 3des-ecb.enc -out 3des-ecb.txt

secure

cat 3des-ecb.txt

Decrypt 3DES CBC Cipher text

To decrypt the 3DES CBC ciphertext, use the following command:

openssl enc -d -des-ede3-cbc -in 3des-cbc.enc -out 3des-cbc.txt

secure

cat 3des-cbc.txt

When prompted for the passphrase, enter:

secure

What type of encryption is the DES algorithm?

Check

PreviousNext

53 Minutes Remaining

kali@kali: ~/Desktop/Challenges/Cryptography/Symmetric

File Actions Edit View Help

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat salted.txt

Salted_E:*****c*2*1w

Salted_D:*****c*****aB,*(

Salted_R:***02e/49*ecce

Salted_I:***2ec/49*ecce

Salted_***A*2*E*0X*****I*

Salted_***1yL*****j***

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-ecb -in des-ecb.enc -out des-ecb.txt

enter DES-ECB decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat des-ecb.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-cbc -in des-cbc.enc -out des-cbc.txt

enter DES-CBC decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat des-cbc.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-ede3-ecb -in 3des-ecb.en c -out 3des-ecb.txt

enter DES-ECB decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat 3des-ecb.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$

© 2025 Infosec Institute, Inc. Infosec, a Cengage Group company.

PrivacyTerms & Conditions

Figure 4. Decrypting 3DES-ECB ciphertext with OpenSSL.

Lab Assignment 8

app.infosecinstitute.com

Back to playlist

Security+ - Cryptography

Lab · 1 hour

INFOSEC

Exit Lab

Instructions

Resources

openssl enc -d -des-cbc -in des-cbc.enc -out des-cbc.txt

When prompted for the password, enter:

secure

Therefore, to view the content, use the following command:

cat des-cbc.txt

Decrypt 3DES ECB Cipher text

The command below decrypts the 3DES ECB ciphertext:

openssl enc -d -des-ede3-ecb -in 3des-ecb.enc -out 3des-ecb.txt

secure

cat 3des-ecb.txt

Decrypt 3DES CBC Cipher text

To decrypt the 3DES CBC ciphertext, use the following command:

openssl enc -d -des-ede3-cbc -in 3des-cbc.enc -out 3des-cbc.txt

When prompted for the passphrase, enter:

secure

cat 3des-cbc.txt

What type of encryption is the DES algorithm?

Check

Previous

Next

52 Minutes Remaining

kali@kali: ~/Desktop/Challenges/Cryptography/Symmetric

File Actions Edit View Help

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-ecb -in des-ecb.enc -out des-ecb.txt

enter DES-ECB decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat des-ecb.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-cbc -in des-cbc.enc -out des-cbc.txt

enter DES-CBC decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat des-cbc.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-ede3-ecb -in 3des-ecb.enc -out 3des-ecb.txt

enter DES-EDE3-ECB decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat 3des-ecb.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ openssl enc -d -des-ede3-cbc -in 3des-cbc.enc -out 3des-cbc.txt

enter DES-EDE3-CBC decryption password:

*** WARNING : deprecated key derivation used.

Using -iter or -pbkdf2 would be better.

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$ cat 3des-cbc.txt

Cryptography

kali@kali:~/Desktop/Challenges/Cryptography/Symmetric\$

© 2025 Infosec Institute, Inc. Infosec, a Cengage Group company.

Privacy

Terms & Conditions

Figure 5. Decrypting 3DES-CBC ciphertext with OpenSSL.

Lab Assignment 8app.infosecinstitute.com

INFOSEC SkillsLearnRolesTeamsNavigatorBeta

Back to playlist

Security+ - AAA

Lab · 1 hour

Kali Linux - Student

03:47 PM

kali@kali: ~

FileActionsEditViewHelp

kali@kali:~\$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa): ssh-keygen -t rsa -b 4096
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ssh-keygen -t rsa -b 4096
Your public key has been saved in ssh-keygen -t rsa -b 4096.pub
The key fingerprint is:
SHA256:Efcx42/JwEhIG+Rjz6NoV1gErNek5fAdUuj6XTkACKY kali@kali
The key's randomart image is:
+--[RSA 4096]--+
o+Bo+o+
o o+O+o+.
E .+Xo+o+.
..oB= o+.
..S.= .+.
..o .+.
o o. .+.
+--[SHA256]--+
kali@kali:~\$

INFOSEC

Exit Lab

InstructionsResources

Security+ - AAA

Authentication Management - Password Keys

Secure Shell (SSH) is a cryptographic network protocol that uses public and private keys for authentication. Users can generate key pairs and log into servers that support key-based authentication. The server uses the correct public key to verify the client's private key, and if they match, the user is authenticated.

The command below generates RSA key pairs with a 4096-bit length. By default, the keys are stored in the user's home directory, but another directory can be specified.

If a passphrase is added, the private key is encrypted. The user must decrypt it with the passphrase when authenticating. The encrypted key can only be brute-forced if an attacker obtains the key itself.

ssh-keygen -t rsa -b 4096

Generate keys

An SSH connection can be established between the host machine and the container. To enable communication using this protocol, both devices must authenticate each other using key pairs.

To begin, switch to the Student machine and open a new terminal and repeat the steps for creating key pairs:

ssh-keygen -t rsa -b 4096

On your device, the path /home/kali/.ssh will be set to the .ssh folder on your home directory.

PreviousNext

55 Minutes Remaining

Figure 6. Creating an SSH keypair.

INFOSEC Skills Learn Roles Teams Navigator Beta

← Back to playlist

Security+ - AAA

Lab · 1 hour

Kali Linux - Student

root@labuser-virtual-machine: ~

```
File Actions Edit View Help

kali@kali:~$ ssh-copy-id root@192.168.1.100
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/kali/.ssh/id_rsa.pub"
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be established.
ED25519 key fingerprint is SHA256:mskd5GrLL0QuY562UpbPJLxCO26mNSFEAn1AcP9zd0Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@192.168.1.100's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@192.168.1.100'"
and check to make sure that only the key(s) you wanted were added.

kali@kali:~$ ssh root@192.168.1.100
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

170 updates can be applied immediately.
168 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
root@labuser-virtual-machine:~# whoami
root
root@labuser-virtual-machine:~#
```

INFOSEC

Instructions Resources

Copying Keys and Connecting using Keys

First, make sure ssh is enabled and running on the host. Then copy the public key to the second machine using the ssh-copy-id command. When the host tries to connect to the machine, the host's private key is matched against the public key in the machine, authenticating the host and allowing it access.

Run the following commands on your student machine.

```
ssh-copy-id root@192.168.1.100
```

yes

When prompted for the password, enter:

```
passw0rd!
```

The terminal prompts to confirm if it's safe to connect to the Docker container. This occurs when two devices connect for the first time or their key pairs change. The ssh-copy-id command saves the public key in the user's path and requests validation by asking for the root password. Authorization requires both key pairs and valid credentials. Connect to the ssh server using the private key (run the command on the student machine):

```
ssh root@192.168.1.100
```

To move onto the next step, execute the `whoami` command and save the result to a test file.

```
whoami
```

What output will be displayed?

```
u
```

← Previous Next →

48 Minutes Remaining

Figure 7. Connecting to 192.168.1.100 with the SSH key generated in Figure 6.

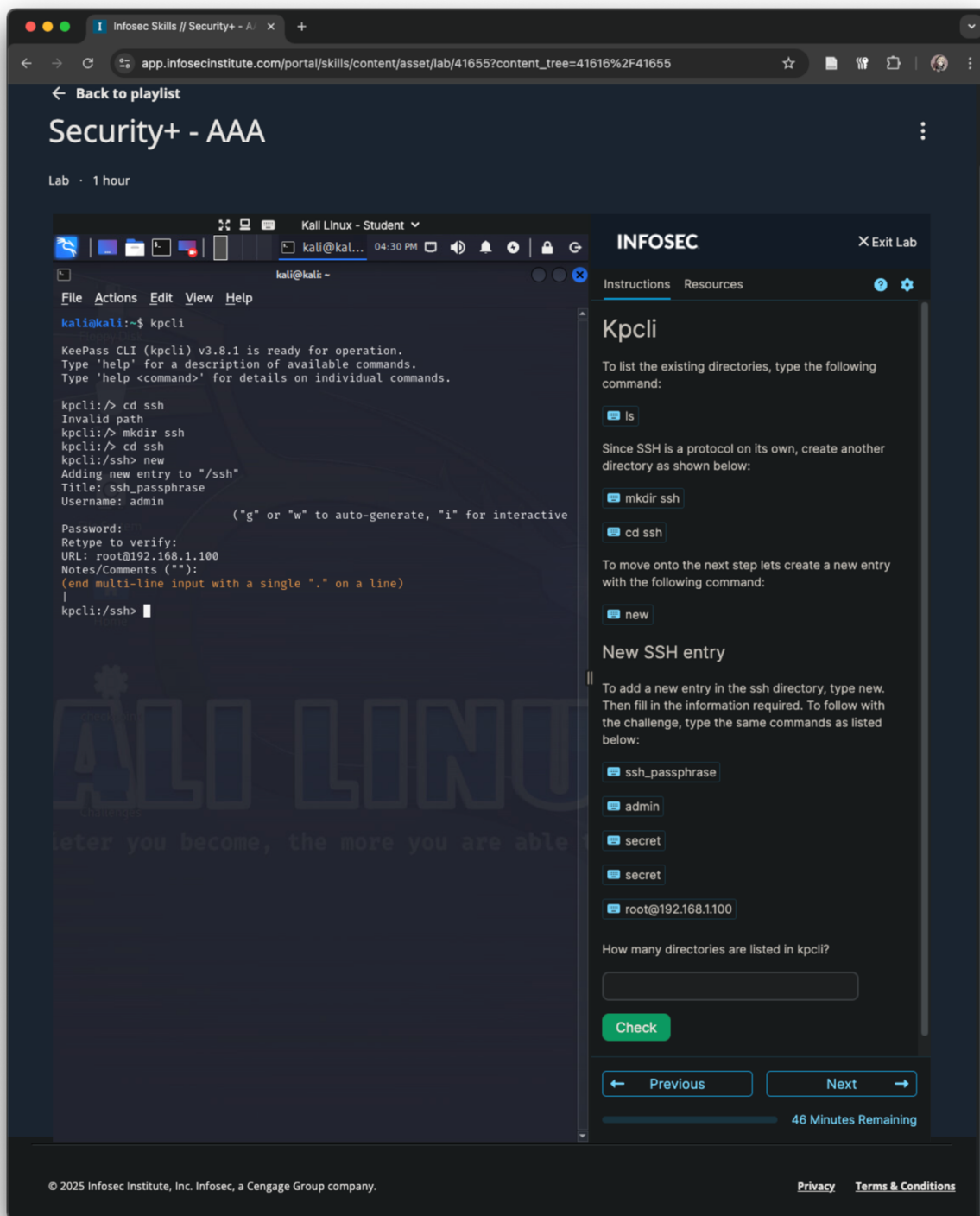


Figure 8. Adding SSH credentials to KeePass.

Infosec Skills // Security+ - AAA

app.infosecinstitute.com/portal/skills/content/asset/lab/41655?content_tree=41616%2F41655

Back to playlist

Security+ - AAA

Lab · 1 hour

Kali Linux - Student

kali@kali: ~

File Actions Edit View Help

kali@kali:~\$ kpcli

KeePass CLI (kpcli) v3.8.1 is ready for operation.
Type 'help' for a description of available commands.
Type 'help <command>' for details on individual commands.

kpcli:/> cd ssh

Invalid path

kpcli:/> mkdir ssh

kpcli:/> cd ssh

kpcli:/ssh> new

Adding new entry to "/ssh"

Title: ssh_passphrase

Username: admin

("g" or "w" to auto-generate, "i" for interactive

Password:

Retype to verify:

URL: root@192.168.1.100

Notes/Comments (""):

(end multi-line input with a single "." on a line)

|

kpcli:/ssh> .

.: unknown command

kpcli:/ssh> saveas ssh_credentials.kdbx

Provide the master password: *****

Retype to verify: *****

You are now operating on file: ssh_credentials.kdbx

kpcli:/> cd ssh/

kpcli:/ssh> ls

Entries

0. ssh_passphrase root@192.168.1.100

kpcli:/ssh> show 0

Title: ssh_passphrase

Uname: admin

Pass:

URL: root@192.168.1.100

Notes:

kpcli:/ssh> show 0 -f

Title: ssh_passphrase

Uname: admin

Pass: secret

URL: root@192.168.1.100

Notes:

kpcli:/ssh>

INFOSEC

Exit Lab

Instructions Resources

saveas ssh_credentials.kdbx

When prompted for the master password, enter the following password:

secret

The master password is used to encrypt and decrypt the database. When choosing a master password, make sure to select a secure and robust password.

While not necessary since we haven't left kpcli, if you were to exit and try to get back in you would have to enter the password. If that were the case you would start kpcli again, load the database and type the new password:

open ssh_credentials.kdbx

secret

Viewing SSH Credentials

Change directory to ssh if you are not already there and list the content to see the password entries:

cd ssh/

ls

The show command lists all the details.

show 0

show 0 -f

Which flag is used to see the password in plaintext?

Check

Previous Next

45 Minutes Remaining

© 2025 Infosec Institute, Inc. Infosec, a Cengage Group company.

Privacy Terms & Conditions

Figure 9. Saving and viewing the changes made to the KeePass database in Figure 8.

Infosec Skills // Security+ - AAA

app.infosecinstitute.com/portal/skills/content/asset/lab/41655?content_tree=41616%2F41655

← Back to playlist

Security+ - AAA

Lab · 1 hour

Kali Linux - Student

kali@kali: ~

File Actions Edit View Help

```
kpcli:/> cd ssh
kpcli:/ssh> new
Adding new entry to "/ssh"
Title: ssh_passphrase
Username: admin

Password:
Retype to verify:
URL: root@192.168.1.100
Notes/Comments (""):
(end multi-line input with a single "." on a line)
|
kpcli:/ssh> .
.: unknown command
kpcli:/ssh> saveas ssh_credentials.kdbx
Provide the master password: *****
Retype to verify: *****
You are now operating on file: ssh_credentials.kdbx
kpcli:/> cd ssh/
kpcli:/ssh> ls
== Entries ==
0. ssh_passphrase                                root@192.168.1.100
kpcli:/ssh> show 0
Title: ssh_passphrase
Uname: admin
Pass: 
URL: root@192.168.1.100
Notes:

kpcli:/ssh> show 0 -f
Title: ssh_passphrase
Uname: admin
Pass: secret
URL: root@192.168.1.100
Notes:

kpcli:/ssh> open /home/kali/Desktop/Challenges/Password_vaults/test_credentials.kdbx
Provide the master password: *****
kpcli:/> ls
== Groups ==
eMail/
Internet/
test/
kpcli:/> cd test/
kpcli:/test> ls
== Entries ==
0. test_password
kpcli:/test> show 0 -f
Title: test_password
Uname: test
Pass: secure_password
URL:
Notes:

kpcli:/test> 
```

INFOSEC

Exit Lab

Instructions Resources

KeePass database

There is another KeePass database located on the admin's Desktop. Running the following commands, displays the password.

- open /home/kali/Desktop/Challenges/Password_vaults/test_credentials.kdbx
- admin
- ls
- cd test/
- ls
- show 0 -f

What is the password displayed in the output?

Check

← Previous Next →

44 Minutes Remaining

© 2025 Infosec Institute, Inc. Infosec, a Cengage Group company. [Privacy](#) [Terms & Conditions](#)

Figure 10. Viewing the test_password entry in the test_credentials KeePass database.