

✓ Terminé : Remettre un travail

Ouvert le : lundi 12 décembre 2022, 08:40

À remettre : lundi 12 décembre 2022, 11:40

EPREUVE FINALE COURS SECURITE LOGICIELLE

FINAL EXAM SOFTWARE SECURITY COURSE

12/12/2022

durée de l'épreuve: 2h15 - tiers temps: 3h

Tout d'abord, notez que vous avez le droit d'accéder au site Moodle du cours Sécurité Logicielle. Vous n'êtes pas supposés consulter d'autre ressource sur Internet sauf si explicitement mentionné ou autorisé (ce qui pourra être considéré comme une fraude). La communication entre étudiants ou avec d'autres personnes est strictement interdite et sera considérée comme fraude. Vos téléphones mobiles doivent être rangés dans vos sacs et les applications de messagerie ou de messagerie instantanée ou réseaux sociaux de toutes sorte doivent être arrêtées.

First of all, please note that you are allowed to access the Software Security course on the Moodle website. You are not supposed to browse other resources on the Internet unless otherwise mentioned or authorized (this might be considered a fraud). Communications between students or with other people is strictly forbidden and will be considered a fraud. Mobile phones should be stored within your bags, and instant messaging applications or social networks of any sort and email should be stopped.

Téléchargez d'abord le fichier test.tgz au bas de cette page à partir de votre VM SEED. Décompressez la ensuite avec `tar xzvf test.tgz` / *First download the SecLog.tgz file at the bottom of this page from your SEED VM. Uncompress it with `tar xzvf test.tgz` within you home directory.*

Si vous avez besoin de créer un répertoire partagé avec votre ordinateur hôte, consultez : / *If you would need to set up the shared directory with your host computer, have a look at:*

<https://github.com/seed-labs/seed-labs/blob/master/manuals/vm/seedvm-manual.md>

Vous devez fournir un rapport sur chaque question + des copies d'écran de ce que vous êtes parvenus à faire pour chaque exercice. Il vous est fortement conseillé de soumettre un fichier zip contenant plusieurs documents. Attention, vous êtes limités à un maximum de 20 documents par Moodle (vous pouvez bien sûr soumettre plusieurs documents dans un même fichier zip ou copier-coller des copies d'écran dans un même rapport pour cette épreuve) ! / *You have to deliver a text report for every question + screenshots of what you could achieve for each exercise. You are strongly advised to submit a zip file containing multiple documents. Beware, you are restricted to a maximum of 20 documents by Moodle (you can of course submit multiple documents within a single zip file or copy-paste several screenshots into your test report)!*

En cas de problème avec Moodle, demandez au surveillant de pouvoir envoyer votre rapport final par mail avant l'heure de fin de l'épreuve. / *In case you have a problem with Moodle, ask the supervisor to let you send your final report by email at the latest before the exam is finished.*

Question 1:

Votre mission, si vous l'acceptez, est de prendre connaissance d'un message caché dans le programme stack1. Vous ne devrez utiliser qu'un buffer overflow pour ce faire car ce programme est déployé sur un serveur web que vous ne contrôlez pas. Vous disposez du code source de ce programme et essayez de cracker le programme sur votre machine avant d'attaquer le serveur. / *Your mission, should you accept it, is to discover the message hidden in program stack1. You should only use a buffer overflow to do so for this program is deployed on a web server you don't control. The source code of this program is available to you and you are trying to crack the program on your computer before attacking the server.*

Tous les fichiers nécessaires sont disponibles sous le répertoire `~/test/Overflow`. Le programme stack1 a été compilé sans les mécanismes de sécurité normalement nécessaires et en 32 bits. Vous avez aussi accès au code source du programme, vidé de son message secret. Vous pouvez le compiler et le faire tourner sous gdb si vous le souhaitez. / *All the necessary files are available under the `~/test/Overflow` directory.*

The stack1 program has been compiled without proper security features, and in 32 bits. You also have access to the source code of the program, stripped from its secret message. You may compile it and run it with gdb if you wish so.

TODO:

Vous devez générer un fichier badfile qui sera utilisé pour effectuer l'attaque et révéler le message. Vous pouvez modifier le programme exploit.py qui était utilisé lors du TD ou créer votre propre générateur, éventuellement dans un autre langage (Indice: la structure de badfile n'est pas nécessairement la même que dans le TD, prenez un peu de temps pour analyser le programme vulnérable). / *You have to generate a badfile that will be used to execute the attack and reveal the message. You might modify the exploit.py program that was used in the lab or create your own generator, possibly in another language (Hint: the structure of the badfile is not necessarily the same as in the lab, take some time to analyze the vulnerable program!)*

Fournissez le code du générateur, le message caché, une copie d'écran de l'exécution du programme et une explication précise de votre approche. Des points supplémentaires (bonus) pourront être obtenus si vous décrivez plusieurs approches pour l'attaque. / *Provide the code of the generator, the message hidden, a screenshot of the execution of the program and a precise explanation of your approach. Bonus points might be received if you describe multiple attack approaches.*

Si vous en avez vraiment besoin, vous pouvez accéder à un convertisseur d'hexadécimal ici : / *If you really need to, you can access an hexadecimal converter here:*

<https://www.rapidtables.com/convert/number/decimal-to-hex.html>

Voici un récapitulatif des commandes disponibles ou que vous devez utiliser au départ : / *A final reminder about the commands available or that you need to run in the first place:*

- `sudo sysctl -w kernel.randomize_va_space=0`
- `gcc -m32 -o stack1 -z execstack -fno-stack-protector stack1.c`
- `gcc -g` (if you need debugging)
- `./exploit` (you may need to `chmod u+x` the file)
- `stack1`

Question 2:

Vous êtes enquêteur de police et vous savez que des individus louches échangent des informations sur un forum qui permet d'envoyer des messages chiffrés. Vous savez que le forum met en oeuvre un chiffrement AES 128 bits en mode OFB que vous ne pouvez déchiffrer par énumération. / *You are a police investigator and you know that shady individuals are exchanging information over a forum that makes it possible to send encrypted messages. You know that the forum implements an AES 128 bit encryption in OFB mode that you cannot break with brute force.*

Un agent de police infiltré dans la bande arrive à mettre la main sur le texte clair d'un de ces messages chiffrés et vous les fait parvenir (fichiers interception.txt et interception.bin). D'après les messages chiffrés (*.bin) que vous voyez dans le répertoire ~/test/Crypto/, cette information vous permet-elle de déchiffrer les autres messages ? Si oui, quel est leur texte (expliquez comment vous procédez) ? Sinon, pourquoi est-ce impossible ? / *An undercover agent in this group manages to snap the cleartext message corresponding to one such encrypted messages and sends them to you (files interception.txt and interception.bin). According to the encrypted messages (*.bin) that you can see in directory ~/test/Crypto/, does this information make it possible to decrypt the other messages? If so what is their content (explain how you proceed)? If not, why is it impossible?*

Suggestions:

Vous pouvez consulter le contenu des fichiers binaires soit en mode texte avec la commande hexdump, soit en mode graphique avec l'utilitaire bless. / *You can browse the content of binary files in text mode with the hexdump command or in graphical mode using the bless utility.*

Question 3:

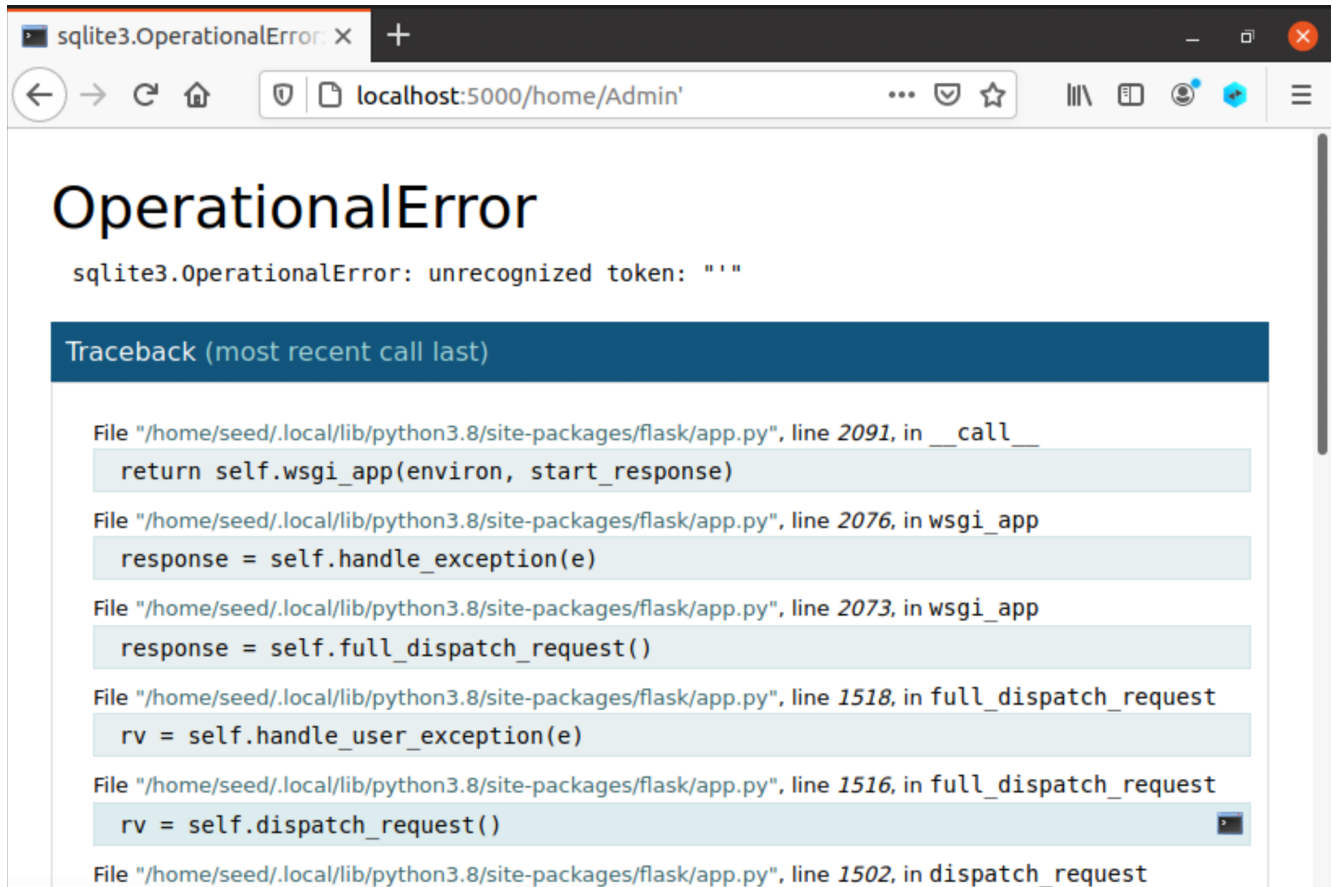
Fonctionnement : / Operation:

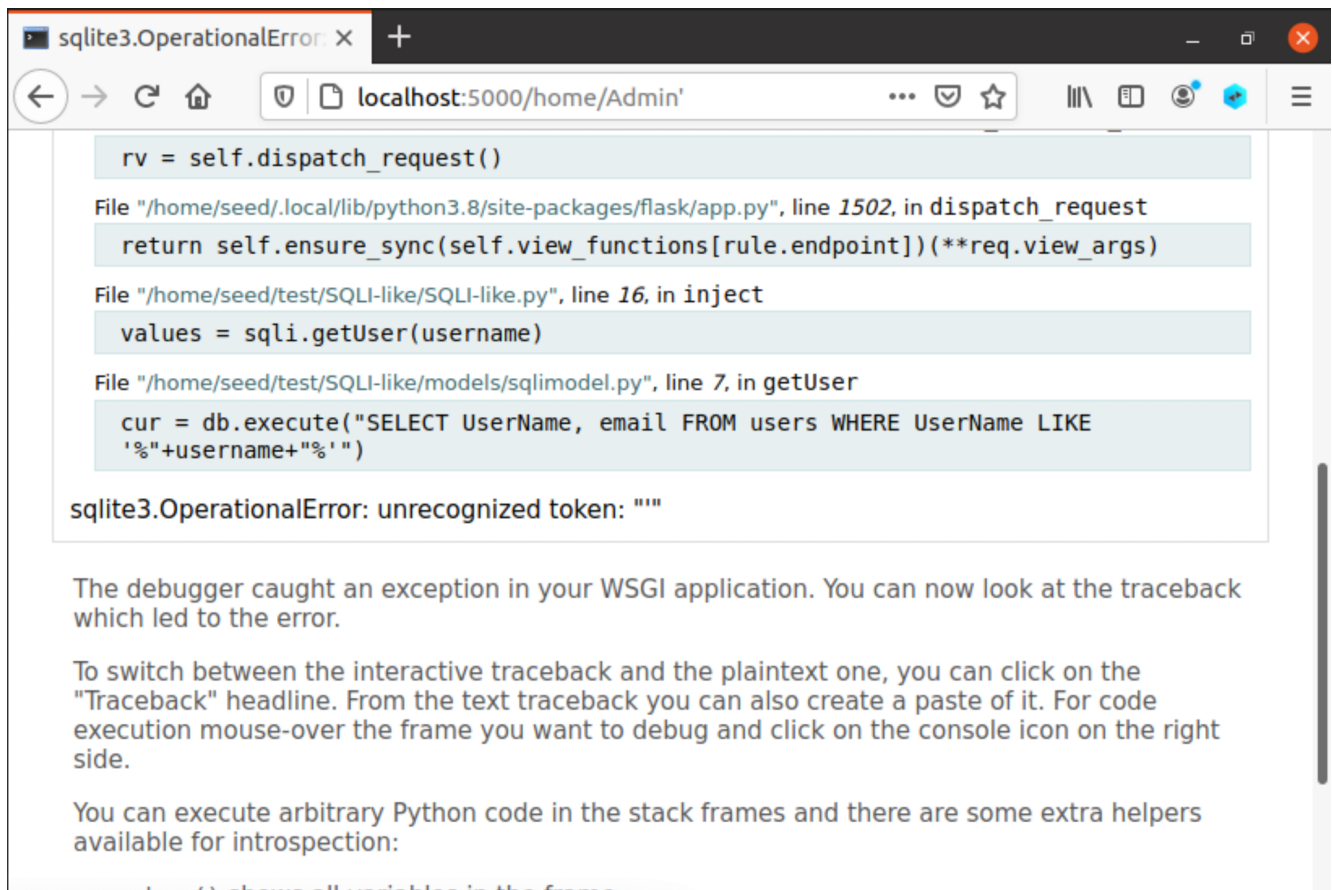
- `cd ~/test/SQLI-like/`
- `pip3 install -r requirements.txt`
- `pip3 install Flask==2.0.3`
- `pip3 install Jinja2==3.1.1`
- `python3 SQLI-like.py`

- Firefox: web page @ <http://localhost:5000/>
- (CTRL-C pour arrêter le serveur si besoin / *to stop server if needed*)

TODO:

- Comme vous avez suivi le cours Sécurité Logicielle à Polytech, vous vous êtes mis en tête de vérifier si l'application web que vous venez de lancer comporte une injection SQL. Vous entrez l'URL « <http://localhost:5000/home/Admin> ». Pourquoi ? / Since you have been following the Software Security course at Polytech, you have decided to test whether the web application that you just launched is vulnerable to SQL injections. You enter URL « <http://localhost:5000/home/Admin> ». Why?
- Expliquez le résultat obtenu sur les captures d'écran suivantes et corrigez l'URL pour éviter ces erreurs. *Explain the result that you get on the following screenshots and change the URL to avoid these errors.*





```
rv = self.dispatch_request()

File "/home/seed/.local/lib/python3.8/site-packages/flask/app.py", line 1502, in dispatch_request
    return self.ensure_sync(self.view_functions[rule.endpoint])(**req.view_args)

File "/home/seed/test/SQLi-like/SQLi-like.py", line 16, in inject
    values = sqli.getUser(username)

File "/home/seed/test/SQLi-like/models/sqlmodel.py", line 7, in getUser
    cur = db.execute("SELECT UserName, email FROM users WHERE UserName LIKE
'%"+username+"%'")

sqlite3.OperationalError: unrecognized token: ""
```

The debugger caught an exception in your WSGI application. You can now look at the traceback which led to the error.

To switch between the interactive traceback and the plaintext one, you can click on the "Traceback" headline. From the text traceback you can also create a paste of it. For code execution mouse-over the frame you want to debug and click on the console icon on the right side.

You can execute arbitrary Python code in the stack frames and there are some extra helpers available for introspection:

`dump()` shows all variables in the frame

- Fournissez la liste des utilisateurs enregistrés dans la base de données en expliquant comment vous procédez (commandes et captures d'écran nécessaires). / Provide the list of users registered in the database and explain how you proceed (commands and screenshots are mandatory).
- Expliquez comment vous pouvez récupérer l'ensemble des informations à propos des utilisateurs (et donnez en le contenu). Y a-t-il un risque supplémentaire ? / Explain how you can retrieve all the information about the different users (and provide this content). Is there any additional risk incurred?
- Pouvez-vous altérer des informations à propos des utilisateurs enregistrés dans la base de données ? / Can you alter information about the users registered in the database?

Suggestions:

- #1: La base de données utilisée est SQLite (qui utilise -- pour les commentaires au lieu de %) / The database used is SQLite (which uses -- for comments instead of %).
- #2: le schéma de la base peut être récupéré à partir de sqlite_master dans SQLite (voir documents) / the database schema can be retrieved from the sqlite_master in SQLite (see documents)
- #3: Les messages d'erreur fournissent aussi des informations utiles / Error messages also provide useful information

Notes et Ressources / Notes and Resources available:

Vous trouverez plusieurs documents sous le répertoire ~/test/SQLi-like/Docs / You will find several documents under the ~/test/SQLi-like/Docs directory:

- Une documentation de SQLite / a documentation of SQLite (simpleSQLite Documentation.pdf)
- Une documentation sur le hacking sur SQLite / a documentation about hacking into SQLite (41397-injecting-sqlite-database-based-applications.pdf)
- Deux cours SQL pour référence / Two SQL courses for your reference (c4.pdf, cbd-sql.pdf).

 [test.tgz](#)

11 décembre 2022, 17:14

Modifier le travail

Supprimer travail remis

Statut de remise

Numéro de tentative

Ceci est la tentative 1 (2 tentatives permises).

Statut des travaux remis	Remis pour évaluation									
Statut de l'évaluation	Non évalué									
Temps restant	Le travail a été remis en avance de 15 min 32 s									
Dernière modification	lundi 12 décembre 2022, 11:24									
Remises de fichiers	<table><tr><td></td><td>decrypt.py</td><td>12 décembre 2022, 11:24</td></tr><tr><td></td><td>exploit.py</td><td>12 décembre 2022, 11:24</td></tr><tr><td></td><td>Florian Latapie.pdf</td><td>12 décembre 2022, 11:24</td></tr></table>		decrypt.py	12 décembre 2022, 11:24		exploit.py	12 décembre 2022, 11:24		Florian Latapie.pdf	12 décembre 2022, 11:24
	decrypt.py	12 décembre 2022, 11:24								
	exploit.py	12 décembre 2022, 11:24								
	Florian Latapie.pdf	12 décembre 2022, 11:24								
Commentaires	<div>► Commentaires (0).</div>									

✉ [Contacter l'assistance du site](#) ↗

Connecté sous le nom « [latapie Florian](#) » ([Déconnexion](#))

[Résumé de conservation de données](#)

[Obtenir l'app mobile](#)

Fourni par [Moodle](#)