

# SciAuth Student Fellow: Final Presentation

December 12, 2022

Abdullah Al Raqibul Islam  
PhD Student, UNC Charlotte

SciAuth student fellow, Fall - 2022  
Mentor: Dr. Derek Weitzel

# Outline

- Project goals
- Project outcomes
- Learning opportunities and future direction

# Project goals

- Build a python based compliance test suite to verify SciTokens

# Project goals

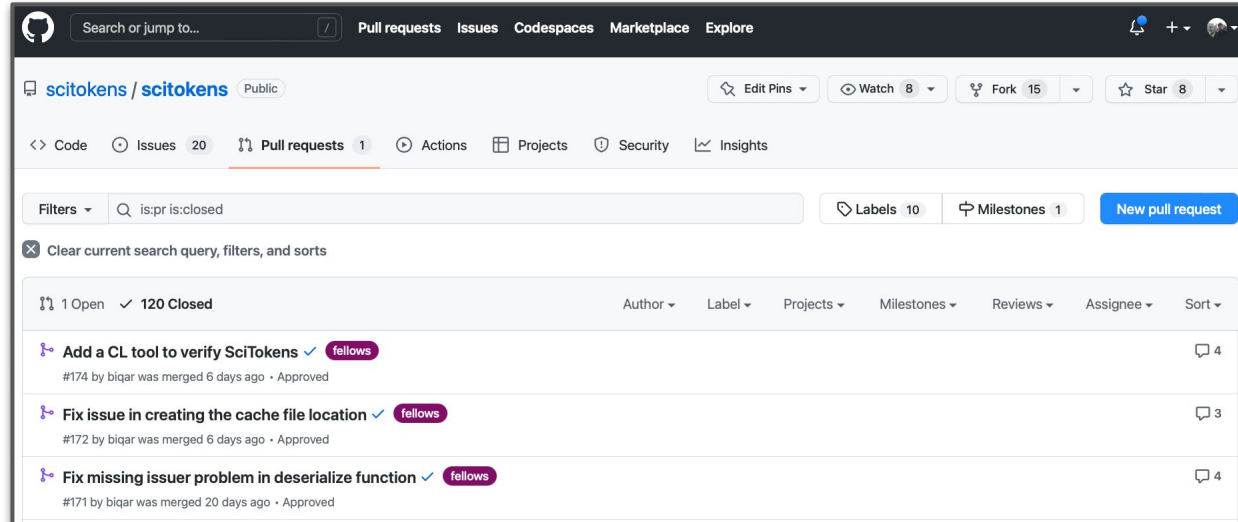
- Build a python based compliance test suite to verify SciTokens



Figure: Workflow of python based test suite to verify SciTokens

# Project outcomes

- Build a python based compliance test suite to verify SciTokens
- Reported/fixed two bugs

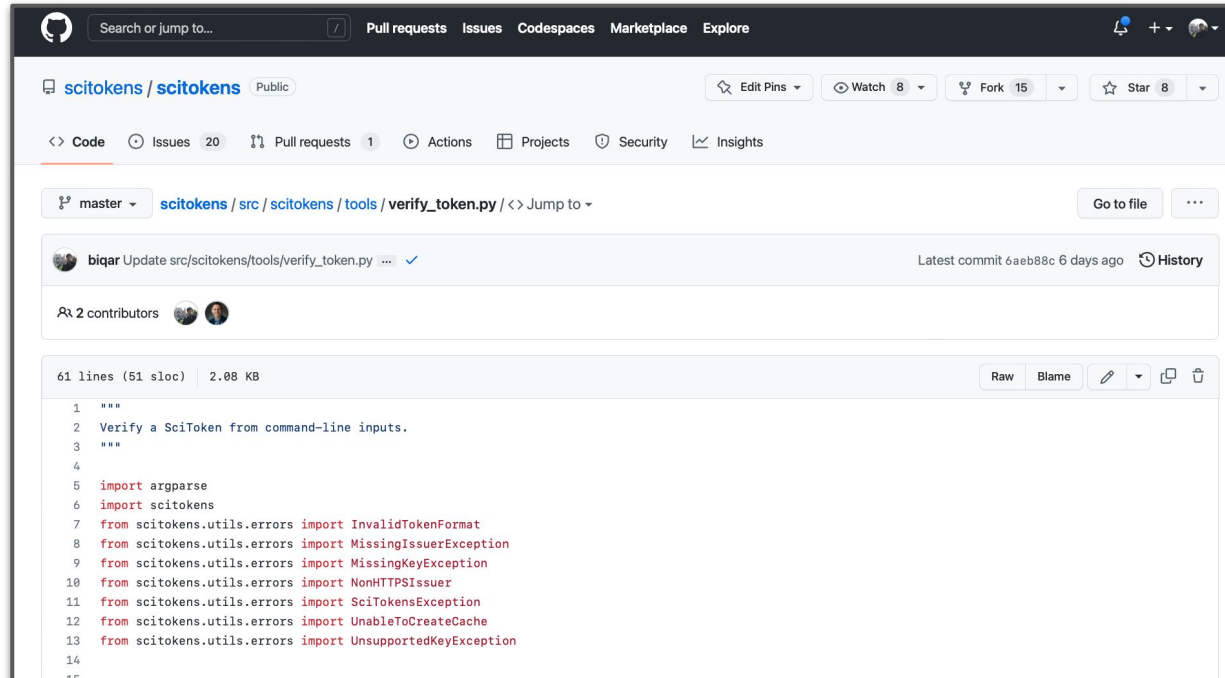


The screenshot shows the GitHub interface for the repository `scitokens/scitokens`. The page is displaying the Pull requests tab, filtered by `is:pr is:closed`. The search results show three closed pull requests, all of which were merged and approved. The first pull request, #174, is titled "Add a CL tool to verify SciTokens" and was merged 6 days ago. The second pull request, #172, is titled "Fix issue in creating the cache file location" and was also merged 6 days ago. The third pull request, #171, is titled "Fix missing issuer problem in deserialize function" and was merged 20 days ago. Each pull request entry includes a checkmark indicating it was merged, a "fellows" label, and a comment icon.

Open	Closed	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
1	120							
	<a href="#">Add a CL tool to verify SciTokens</a> ✓ fellows							4
	#174 by biqar was merged 6 days ago · Approved							
	<a href="#">Fix issue in creating the cache file location</a> ✓ fellows							3
	#172 by biqar was merged 6 days ago · Approved							
	<a href="#">Fix missing issuer problem in deserialize function</a> ✓ fellows							4
	#171 by biqar was merged 20 days ago · Approved							

# Project outcomes: verify\_token.py

- Can find it at: `scitokens/src/scitokens/tools`



The screenshot shows a GitHub repository page for the `scitokens/scitokens` project. The file `verify_token.py` is selected, showing its commit history and code content. The code is a Python script for verifying SciTokens from command-line inputs.

```
1 """
2 Verify a SciToken from command-line inputs.
3 """
4
5 import argparse
6 import scitokens
7 from scitokens.utils.errors import InvalidTokenFormat
8 from scitokens.utils.errors import MissingIssuerException
9 from scitokens.utils.errors import MissingKeyException
10 from scitokens.utils.errors import NonHTTPSIssuer
11 from scitokens.utils.errors import SciTokensException
12 from scitokens.utils.errors import UnableToCreateCache
13 from scitokens.utils.errors import UnsupportedKeyException
14
```

# Project outcomes: `verify_token.py`

- Can provide user friendly message for the following type of exceptions:
  - `MissingKeyException`
  - `UnsupportedKeyException`
  - `MissingIssuerException`
  - `NonHTTPSIssuer`
  - `InvalidTokenFormat`
  - `UnableToCreateCache`
  - `SciTokensException`

# Project outcomes: verify\_token.py

- In case of a valid input token, the program will print the claims.

```
raqib-pro:tools aislam6$ python3 verify_token.py "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6ImtleS1yc2I1NiJ9.eyJ2ZXIiOiJzY2l0b2t1bjoyLjAiLCJhdWQiOiJodHRwczovL2RlbW8uc2NpdG9rZW5zLm9yZyIsImZlcyI6Imh0dHBz0i8vZGVtby5zY2l0b2t1bnMub3JnIiwiaXhwIjoxNjcwODY4MDI3LCJpYXQiOjE2NzA4Njc0Mjc5Im5iZiI6MTY3MDg2NzQyNywianRpIjoia0DUzZDc4MmMtNDM0Yy00Y2IzLWFhYmMtN2JiYjU0NDIyZjhlin0.PNQISM0c9V1TC1Bre_s1LX3fhAxScg-LVuhBQQubs89dBMGeEIQbKNy0IYzy_ME6AMP7n78PRLzgruNLws8NgLjRqxJTy08Cjjm67LUTsFORQbgfPQif9DU5See9LseTphIF1uXH8PvNLI31fEeYtGVM0HVtuMHS69FwPtW_cm2AYGdTVss3EWGYYKbiF8RttwQ8bQlupHCq0qnFrLQRg3XwhZDDmkzLqaJ7kmVNNiDoKJ96w41uozv7K0gGfgu1M8AIGNWva_16LnpUWavMe_ftHs1v_9nhGwGcfZNKYV6Lu0GXaGwVK24mmURC8gVuWMzlmM77w2y-s1vKSgbUcw" -v
Claims:
('ver', 'scitoken:2.0')
('aud', 'https://demo.scitokens.org')
('iss', 'https://demo.scitokens.org')
('exp', 1670868027)
('iat', 1670867427)
('nbf', 1670867427)
('jti', '853d782c-434c-4cb3-aabc-7bbb54422f8e')
raqib-pro:tools aislam6$
```



# Project outcomes: verify\_token.py

- In case of an invalid input token, the program will print the user friendly message.

```
raqib-pro:tools aislam6$  
raqib-pro:tools aislam6$ python3 verify_token.py 'eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiJibGFoLnR4dCI6Imh0dHA6Ly9kZW1vLnNjaXRva2Vucy5vcnciLCJleHAiOiJlbnR5MzZyY3MzYyNTUyMjY2NjcyOTA1MSwibmJmIjoxNjY2NzI5MDUxLzQqdGkiOiIzYTFmOGM2NC02NjM2LTQ3N2E+YjBhNC1hNDlmZWE2YjQ4MWMifQ.Km08tUHNLSZ7KU9It7Towd1UX8gmIcufsjQwR20SztK-UeXinrUA3F6BHppAKb2YGPTHYGiyVD84rrHITZZaKHnUdA23lBMCietJ9fS3PgFTJf1K-yC1qyCmGDQONIPSOs6UZ0MuS7Ww6aUw2NGKoF3Se7zjl8f6ieLcusjy8osvc-C4nZdssZif3m2xZdYaLR0Igw41uP5tU_-CPQzFFj2k4tPVUNIJfE0R-TIE2KuLQ1UXZo0Gx8IPBf1bICtwRvU9tJ6aIrcr4or3brGvR2NYgRSDfZmCVH__9Wiz93MVcGdj4KiMGd_kajv0eUCQDMiH3RkeYFko2s1Fc11xg'  
Issuer is not over HTTPS. RFC requires it to be over HTTPS.  
raqib-pro:tools aislam6$
```

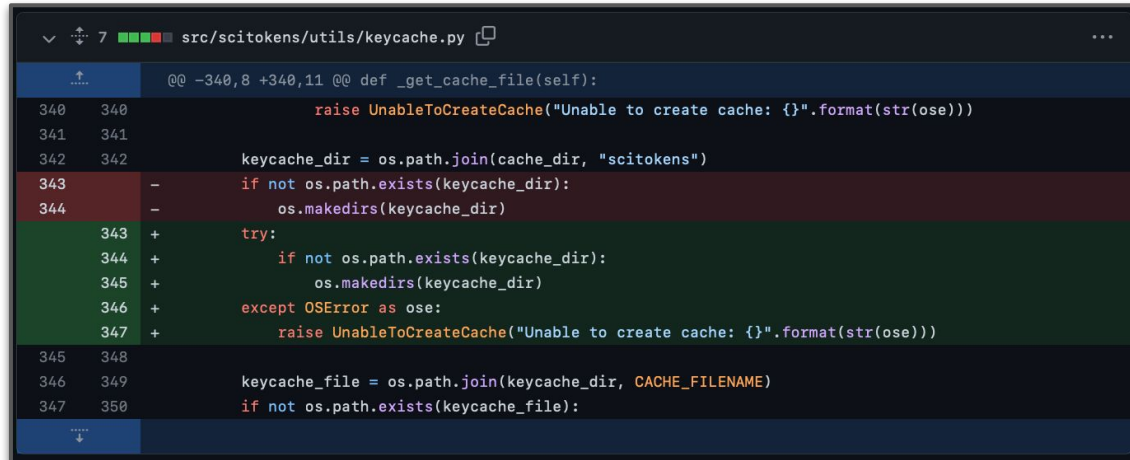
# Project outcomes: Bug Fixes

- **BugFix-1:** The `deserialize()` function in `SciToken` class should raise the “*MissingIssuerException*” in case issuer is not provided.

```
src/scitokens/scitokens.py
@@ -295,6 +295,9 @@ def deserialize(serialized_token, audience=None, require_key=False, insecure=False)
295 295     # Get the public key from the issuer
296 296     keycache = KeyCache.KeyCache().getinstance()
297 297     if public_key == None:
298 +         if 'iss' not in unverified_payload:
299 +             raise MissingIssuerException('Issuer not provided')
300 +
298 301     issuer_public_key = keycache.getkeyinfo(unverified_payload['iss'],
299 302                                             key_id=unverified_headers['kid'] if 'kid' in unverified_headers else None
300 303     insecure=insecure)
```

# Project outcomes: Bug Fixes

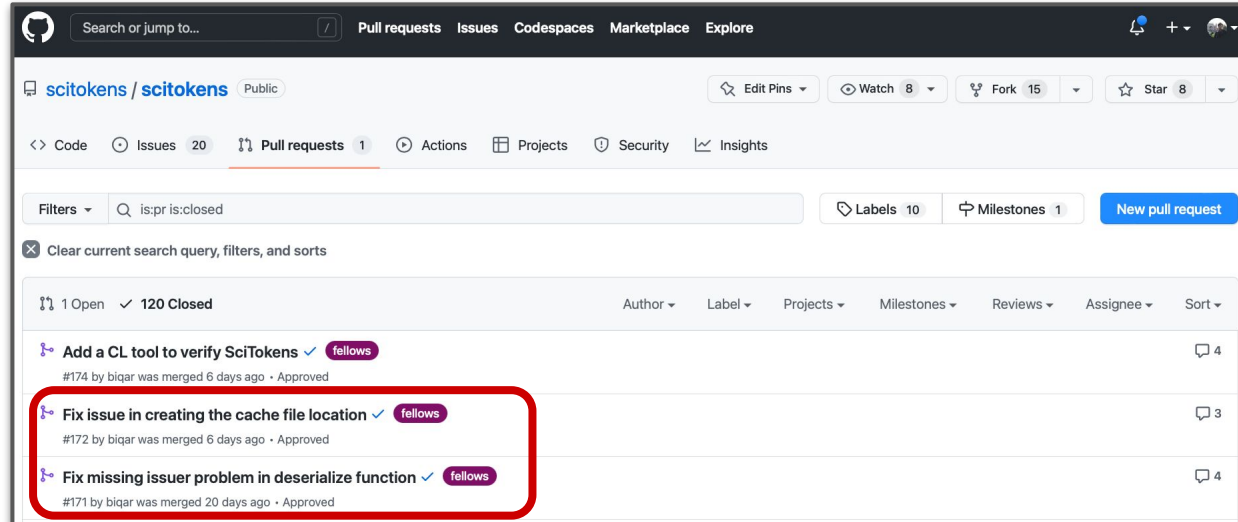
- **BugFix-2:** Cache file directory creation requires to be done under try-catch block.
  - Otherwise important SciToken related exception (e.g., “*UnableToCreateCache*”) was not raised.



```
src/scitokens/utils/keycache.py
@@ -340,8 +340,11 @@ def _get_cache_file(self):
340 340         raise UnableToCreateCache("Unable to create cache: {}".format(str(ose)))
341 341
342 342         keycache_dir = os.path.join(cache_dir, "scitokens")
343 -         if not os.path.exists(keycache_dir):
344 -             os.makedirs(keycache_dir)
343 +         try:
344 +             if not os.path.exists(keycache_dir):
345 +                 os.makedirs(keycache_dir)
346 +             except OSError as ose:
347 +                 raise UnableToCreateCache("Unable to create cache: {}".format(str(ose)))
345 348
346 349         keycache_file = os.path.join(keycache_dir, CACHE_FILENAME)
347 350         if not os.path.exists(keycache_file):
```

# Project outcomes: Bug Fixes

- Finally, added unit tests for these bug fixes.
  - Created and merged two pull requests in the GitHub repo.



The screenshot displays the GitHub interface for the `scitokens/scitokens` repository. The navigation bar includes links for Pull requests, Issues, Codespaces, Marketplace, and Explore. The repository name is shown as `scitokens / scitokens` with a 'Public' label. The 'Pull requests' tab is active, showing a search filter for `is:pr is:closed`. The results list shows three pull requests, all of which are approved and merged:

- Add a CL tool to verify SciTokens** (Approved, Merged 6 days ago, 4 comments)
- Fix issue in creating the cache file location** (Approved, Merged 6 days ago, 3 comments)
- Fix missing issuer problem in deserialize function** (Approved, Merged 20 days ago, 4 comments)

# Learning opportunities and future direction

- Learned about JSON Web Token (JWT) technologies.
- Learned unit-testing in Python.
- Learned about the importance of testing/checking automation in GitHub.
- Future direction: Can develop a web page to allow the users to use the current verification tool over the internet.

# Learning opportunities and future direction

- Learned about JSON Web Token (JWT) technologies.
- Learned unit-testing in Python.
- Learned automa
- Future direction: can develop a web page to allow the users to use the current verification tool over the internet.

Thanks to my mentor  
Dr. Derek Weitzel  
for his kind support!

# Thank You

Question?