هوالحق

تمرین کامپیوتری شماره ۳
مبانی امنیت شبکه، بهار ۱۴۰۳

در این تمرین شما با OAuth 2.0 که یک پروتکل برای Authorization کاربران می‌باشد آشنا خواهید شد. پس از مطالعه کافی در مورد پروتکل OAuth و چگونگی کارکرد آن، دستورالعمل مربوطه را انجام داده، گزارش خود را تهیه کرده و در انتها به سوالات مطرح شده پاسخ دهید. توصیه می‌گردد لینک‌های موجود در بخش منابع را قبل از انجام مراحل دستورالعمل مطالعه کرده و همچنین بخش نکات را پیش از انجام کار مد نظر قرار دهید.

۱) دستورالعمل

در طی این دستورالعمل می‌بایست یک برنامه تحت وب ساده که از سیستم احراز هویت سرویس GitHub.com برای Authorization استفاده می‌کند را تولید نمائید. این برنامه با استفاده از پروتکل OAuth اطلاعات پروفایل کاربر را پس از ورود موفق نمایش خواهد داد. در این تمرین استفاده از روش Authorization Code Grant Type مد نظر می‌باشد.

پروسه Authorization در این روش شامل چهار مرحله خواهد بود:

۱. انتقال کاربر به صفحه احراز هویت سایت GitHub پس از فشردن دکمه ورود در صفحه برنامه ایجاد شده.
۲. ارجاع کاربر پس از احراز هویت توسط GitHub به برنامه ایجاد شده توسط شما و دریافت کد ارجاع.
۳. دریافت Access Token با استفاده از کد ارجاع دریافت شده و پارامترهای کلاینت.
۴. استفاده از Access Token دریافت شده برای درخواست اطلاعات پروفایل کاربر از API سرویس GitHub.

برای پیاده سازی مراحل زیر را طی نمائید:

۱. ابتدا می‌بایست با استفاده از راهنمای موجود در این لینک یک OAuth App در اکانت GitHub خود ایجاد کنید. در فیلد callback URL می‌بایست آدرس ارجاع پس از احراز هویت توسط GitHub را وارد کنید. در این تمرین لینک ارجاع برابر http://localhost:8589/oauth/redirect خواهد بود.

۲. پس از ساختن OAuth App مربوط به خود، فایل سروری که در اختیار شما قرار گرفته شده است (server.py) را با استفاده از دستورات زیر (در مسیر فولدر server) اجرا نمائید. در صورتی که سرور با موفقیت اجرا گردد، پیغامی مبنی بر فعالیت سرور روی پورت 8589 به نمایش در خواهد آمد. همچنین شما می‌توانید به راحتی پورت مربوط به سرور را در فایل server.py تغییر دهید. لازم به ذکر است که در صورت تغییر پورت سرور، مقدار callback URL در OAuth App خود را نیز با توجه به مقدار پورت مربوطه به روزرسانی کنید.

```
$ pip3 install fastapi
$ pip3 install uvicorn
$ python3 server.py
```

هوالحق

تمرین کامپیوتری شماره ۳
مبانی امنیت شبکه، بهار ۱۴۰۳

۳. در این مرحله لازم است تا یک صفحه قابل اجرا و ساده با فرمت HTML و با نام login.html را طراحی کنید. این صفحه می‌بایست شامل یک دکمه با نام Login بوده که با کلیک کاربر بر روی آن به صفحه احراز هویت سایت GitHub منتقل گردد. توجه داشته باشید که درخواست مربوط به انتقال به صفحه احراز هویت GitHub باید شامل پارامترهایی که اطلاعات مربوط به آن‌ها در این لینک موجود است باشد.

۴. کاربر پس از احراز هویت موفقیت آمیز در سایت GitHub به آدرس درج شده در callback URL ارجاع داده می‌شود. همراه با این ارجاع پارامتری با نام code نیز تأمین خواهد شد. سرور server.py با دریافت این ارجاع، پارامتر code را در کنسول در حال اجرا چاپ خواهد کرد و همچنین مقدار آن را در پاسخ برخواهد گرداند.

۵. در این مرحله با در دست داشتن مقدار پارامتر code و همچنین مقادیر client_id و client_secret می‌بایست از سرویس GitHub درخواست Access Token مربوطه را انجام دهید. این درخواست را به صورت دستی و توسط ابزارهای ارسال درخواست HTTP (نظیر cURL یا Postman) ارسال کرده و پاسخ را دریافت کنید. توجه داشته باشید که در گزارش خود حتماً مقادیر مربوطه را ذکر کرده و تصاویر مربوط به انجام مراحل کار را درون گزارش الحاق کنید.

۶. پس از دریافت Access Token امکان استفاده از API سرویس GitHub برای شما مقدور خواهد بود. در این مرحله اطلاعات پروفایل کاربر (مانند نام، ایمیل و غیره) را با استفاده از API دریافت کرده و نمایش دهید. این درخواست را به صورت دستی و توسط ابزارهای ارسال درخواست HTTP ارسال کرده و پاسخ را دریافت کنید. در این بخش نیز مقادیر مربوطه به همراه تصاویر دقیق از فرمت، محتوا و چگونگی مراحل ارسال و دریافت اطلاعات از API را درون گزارش خود ذکر نمائید. این تصاویر می‌بایست حداقل شامل یکبار انجام مراحل برای پروفایل کاربری GitHub خود شما باشد.

۷. در انتها، در تمامی مراحل بالا که درخواست‌های خود را به صورت دستی ارسال می‌کردید، به صورت خودکار و در فایل server.py پیاده سازی کنید. برای اینکار می‌توانید از کتابخانه پایتون Requests استفاده کنید. پیشنهاد می‌شود که پیش از پیاده‌سازی، مطالعه مختصری بر روی مستندات کتابخانه‌های FastAPI و Requests داشته باشید. در این تمرین، از کتابخانه FastAPI برای سرویس‌دهی وب و از کتابخانه Requests برای سرویس‌گیری از وب استفاده می‌کنیم:

[1] https://fastapi.tiangolo.com/
[2] https://docs.python-requests.org/en/latest/

هوالحق

تمرین کامپیوتری شماره ۳
مبانی امنیت شبکه، بهار ۱۴۰۳

۲) سؤالات

۱. مزایای استفاده از روش Authorization Code Grant Type چیست؟

۲. در صورتی استفاده از روش Client Credential Grant Type در یک نرم‌افزار تلفن همراه، چه ضعف(های) امنیتی متوجه این روش خواهد بود؟

۳. آیا نوع Access Token دریافتی از انواع شناخته شده (مانند JWT) می‌باشد؟ آیا می‌توان این Access Token را Decode کرد؟ در مورد نوع و روش تولید این Access Token تحقیق کنید.

۴. با انتقال برنامه تولید شده توسط شما به محیط Production و استفاده از آن در محیط واقعی، حداقل یک مورد ضعف امنیتی برای برنامه شما وجود خواهد داشت. این ضعف امنیتی را شناسایی کرده و برای رفع آن راه حلی را ارائه دهید.

۳) نکات

- توصیه می‌گردد که برای انجام این تمرین از یک توزیع Linux مانند Ubuntu و یا Debian استفاده نمائید.
- لازم به ذکر است که برنامه شما می‌بایست **فقط مجوز دسترسی به اطلاعات پروفایل کاربر را داشته باشد.**
- در هر مرحله از انجام این تمرین، تصاویر مناسب از انجام عملیات را درون گزارش خود الحاق کنید.
- موارد تحویلی شما باید یک فایل فشرده با نام شماره دانشجویی خود، شامل **یک فایل PDF گزارش کار** و همچنین **یک دایرکتوری با نام server** (**حتماً شامل فایل login.html و server.py**) باشد.
- گزارش شما باید لازم و در عین حال کافی باشد. تعداد صفحات یک گزارش کیفیت آن را تعیین نخواهد کرد.
- دقت داشته باشید که در انتهای این تمرین، OAuth App ساخته شده در GitHub را حذف نمائید تا امکان سوء استفاده از آن به وجود نیاید.

۴) منابع

- برای آشنایی با انواع OAuth Grant Type می‌توانید به این لینک مراجعه کنید.
- برای کسب اطلاعات دقیق در مورد API سرویس GitHub از این راهنما استفاده کنید.
- برای آشنایی با مفهوم Scope و انواع آن در پروتکل OAuth سرویس GitHub به این لینک مراجعه کنید.