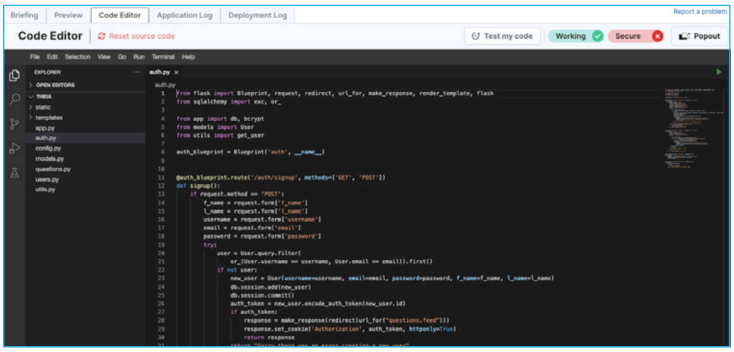
IMMERSIVELABS

DATA SHEET

Application Security and DevSecOps

Business demands are leading to applications being built quicker than ever, while rushed development increases the likelihood of software vulnerabilities. Mitigating application threats needs more than just check-box, multiple-choice training; it requires capabilities that span the entire software development lifecycle (SDLC).

Securing Applications is more than secure coding!
Immersive Labs measures and enhances development teams' security capabilities across the entire SDLC. Hands-on content experiences use real code in real applications, allowing developers to gain practical cybersecurity experience.



```python
from flask import Blueprint, request, redirect, url_for, make_response, render_template, flash
from sqlalchemy import exc, or_

from app import db, bcrypt
from models import User
from utils import get_user

auth_blueprint = Blueprint('auth', __name__)


@auth_blueprint.route('/auth/signup', methods=['GET', 'POST'])
def signup():
    if request.method == 'POST':
        f_name = request.form['f_name']
        l_name = request.form['l_name']
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        try:
            user = User.query.filter(
                or_(User.username == username, User.email == email)).first()
            if not user:
                new_user = User(username=username, email=email, password=password, f_name=f_name, l_name=l_name)
                db.session.add(new_user)
                db.session.commit()
                auth_token = new_user.encode_auth_token(new_user.id)
                if auth_token:
                    response = make_response(redirect(url_for("questions.feed")))
                    response.set_cookie('Authorization', auth_token, httponly=True)
                    return response
                return "Sorry there was an error creating a new user"
```

Benefits

- Developers are able to quickly identify and fix vulnerabilities, increasing productivity
- Hire and upskill junior developers and DevSecOps professionals to reduce hiring costs
- Save money by identifying vulnerabilities earlier in the SDLC
- Address compliance mandates related to secure coding

Audiences

- Software Developers
- Application Security Experts
- DevSecOps professionals
- QA Testers

Learn Through Doing

Interact and fix code in real applications, ensuring your ability to retain functionality in the application. Labs are provided in a sandbox environment, enabling safe practicing and exploration.

See the Attacker's Point of View

Our labs mimic how attackers would typically exploit vulnerabilities, giving your teams an understanding of why secure coding practices are so critical across the SDLC.

Secure the Entire SDLC

Our labs focus on the entire lifecycle of an application, upskilling everyone from QA testers to engineers. We use real-world examples to make developers aware of the actual financial and reputational impact vulnerabilities can have.

Proof of Capability

Leverage data insights to measure and map the maturity of your organization's Engineering, AppSec, and DevSecOps teams over time. With this, teams can prove their capability in numerous ways and identify weak points for improvement.

Trusted by the world's largest companies, governments, and defense organizations

HSBC Pfizer citi Humana McLaren nationalgrid

Move Beyond Multiple-Choice AppSec Training to Hands-On Experiences

At Immersive Labs, we understand that the best way to learn is through doing. A good developer learns from their mistakes, but a great developer also learns from the mistakes of others.

In order to learn, we provide teams with a wide range of common security errors in code and configurations to identify and fix. By offering hands-on AppSec experiences, your teams' knowledge, skills, and judgement on secure coding, secure operations and secure testing will improve.

With these hands-on experiences, developers will:

- See first-hand how attackers exploit vulnerabilities and the impact they pose
- Fix vulnerabilities in a way that retains the application's functionality
- Experiment by modifying code or configurations, observing the impact on exploit attempts, and impact on application functionality

Things you won't do:

- Complete endless multiple-choice questions which are tedious and ineffective learning methods
- Be spoon-fed answers. Our content is created to develop and enhance your understanding and skills - challenging you is a part of that!
- Ultimately, this creates a far more skilled, confident, and productive team, that can prove their competency and resilience.

Supported Languages and Frameworks:

- Python*
- Java*
- Java Spring
- JavaScript Frontends:
  Vue.js
  Angular
  React
- Node.js*
- TypeScript*
- C#*
- C++
- Go
- PHP
- Ruby on Rails

**These languages also feature API specific series*

Key AppSec Categories and Collections:

- OWASP Top 10*
- CWE 25*
- Secure Fundamentals
- TLS Fundamentals
- Secure Coding
- Secure Testing
- Secure Operations
- Secure Engineering
- Secure Headers
- Introduction to Content Security Policy (CSP)
- API Security Collections

**Labs tagged and referenced to this framework*

Certification & Compliance

Trusted by the world's largest organizations

Over **400** customers

>3.5M total labs completed

>100,000 unique users

>1,700 hands-on challenges

Let's get started! Ready to accelerate your Cyber Workforce Resilience journey with stress test exercises? Contact your Immersive Labs Account Manager to learn more.