Tuition delivered by BCET UK
https://bcet.uk

Centre for Applied Research in Software & IT (CARSIT)
R&D and Capstone Project conducted by CARSIT UK
https://carsit.bcet.uk

Endorsed by ISM Education, an Ofqual-regulated UK Awarding Body
https://ismeducation.com



01 / PROGRAMME IDENTITY

Your gateway to application security

A concentrated, practical pathway from core computing foundations to ethical vulnerability assessment and secure remediation.

12
Weekly modules

120
Indicative learning hours

60–70%
Practical emphasis

Award title	Professional Certificate in Ethical Hacking & Secure Programming
Subtitle	Offensive & Defensive Application Security with SecureLab
Duration	3 months / 12 weeks
Learning time	120 indicative hours; approximately 10 hours per week
Delivery	Tutor-led teaching, demos, guided labs, independent practice and capstone
Tuition / R&D	BCET UK / CARSIT UK
Endorsement	ISM Education, an Ofqual-regulated UK Awarding Body

THE SECURELAB METHOD

Security concept → scenario → vulnerable code → authorised attack → evidence → root cause → secure implementation → retest → verification.

02 / 12-WEEK CURRICULUM

The 12-week learning journey

A customised timeline: foundations → core/offensive testing → defensive engineering → capstone.

01 **Python Programming Fundamentals for Cybersecurity**
Validation, files, exceptions and automation

02 **Flask Web Application Development Fundamentals**
Routes, forms, sessions and attack surfaces

03 **Computer Networking & Network Security Fundamentals**
TCP/IP, ports, DNS, HTTP/HTTPS and firewalls

04 **Database Design, SQL & Secure Database Implementation**
ERD, CRUD, least privilege and parameterisation

05 **Cybersecurity & Application Security Fundamentals**
CIA, risks, threats and attack surfaces

06 **Ethical Hacking & Vulnerability Assessment**
Scope, methodology, evidence and reporting

07 **Injection & Client-Side Attacks**
SQL injection, XSS and unsafe input/output

08 **Authentication, Sessions & Access Control**
IDOR, RBAC and session protection

09 **Web & Server-Side Attacks**
CSRF, SSRF, redirects and request manipulation

10 **File, Configuration & Web Security**
Traversal, uploads, headers and hardening

11 **Secure Programming & Secure SDLC**
Secure patterns, secrets, logging and verification

12 **SecureLab Attack–Defend Capstone**
Find → Exploit → Fix → Retest → Report

03 / MODULE GUIDE — PART 1

Technical Foundations I

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 01 / WEEK 1

Python Programming Fundamentals for

AIM

Develop Python skills to understand vulnerable/secure code and write simple security utilities.

KEY CONTENT

- Variables, data types, conditions, loops and functions
- Strings, collections, files and modules
- Exceptions and defensive error handling
- Input validation and security automation

SECURELAB PRACTICAL

Build small utilities for log processing, validation and basic security checks.

LEARNING OUTCOMES

Write simple programs
Validate untrusted input
Handle files/errors safely

EVIDENCE
Python practical tasks

MODULE 02 / WEEK 2

Flask Web Application Development Fundamentals

AIM

Understand how a small web application is structured before attempting to attack or secure it.

KEY CONTENT

- Routes, URL parameters and templates
- GET/POST, forms, request/response flow
- Sessions, cookies and basic authentication
- Database connectivity and error handling

SECURELAB PRACTICAL

Build a mini Flask app, map its inputs and compare normal versus malicious request flow.

LEARNING OUTCOMES

Build routes and forms
Explain client/server flow
Recognise application attack surfaces

EVIDENCE
Flask mini-application

03 / MODULE GUIDE — PART 2

Technical Foundations II

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 03 / WEEK 3
Computer Networking & Network Security

AIM

Understand web communication, exposed services and security controls.

KEY CONTENT

- Client/server and TCP/IP concepts
- IPv4, MAC, TCP/UDP and ports
- DNS, DHCP, HTTP and HTTPS
- Routers, switches, firewalls and reconnaissance

SECURELAB PRACTICAL

Map application network paths and explain protocol/port choices.

LEARNING OUTCOMES

Interpret network communication
Relate ports to attack surfaces
Explain HTTP vs HTTPS

EVIDENCE
Networking worksheet/lab

MODULE 04 / WEEK 4
Database Design, SQL & Secure Database

AIM

Connect relational database design and SQL to security and injection prevention.

KEY CONTENT

- ERD, entities, keys and normalisation
- CRUD: SELECT, WHERE, JOIN, GROUP BY, ORDER BY
- Database accounts and least privilege
- Parameterised queries and secure connectivity

SECURELAB PRACTICAL

Connect Flask to a database; compare unsafe SQL construction with parameterised queries.

LEARNING OUTCOMES

Design a simple database
Use SQL safely
Explain SQLi root cause

EVIDENCE
Secure database practical

03 / MODULE GUIDE — PART 3

Security Mindset & Ethical Assessment

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 05 / WEEK 5

Cybersecurity & Application Security Fundamentals

AIM

Establish security-risk vocabulary and an application-security mindset.

KEY CONTENT

- CIA triad and security objectives
- Assets, threats, vulnerabilities, controls and risk
- Attack surfaces and trust boundaries
- Security-by-design; OWASP/CWE awareness

SECURELAB PRACTICAL

Create a threat and attack-surface map for a SecureLab application.

LEARNING OUTCOMES

Use terminology correctly
Distinguish vulnerability from risk
Identify attack surfaces

EVIDENCE
Threat/attack-surface brief

MODULE 06 / WEEK 6

Ethical Hacking & Vulnerability Assessment

AIM

Teach a controlled, authorised process for validating and reporting weaknesses.

KEY CONTENT

- Authorisation, scope and rules of engagement
- Reconnaissance and enumeration concepts
- Vulnerability identification and validation
- Evidence, severity, reporting and retesting

SECURELAB PRACTICAL

Conduct a guided assessment of an intentionally vulnerable SecureLab target.

LEARNING OUTCOMES

Work within scope
Capture defensible evidence
Write a structured finding

EVIDENCE
Vulnerability finding report

03 / MODULE GUIDE — PART 4

Injection, Identity & Access

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 07 / WEEK 7

Injection & Client-Side Attacks

AIM

Show how unsafe input/output becomes exploitable and how secure code blocks it.

KEY CONTENT

- SQL injection
- Cross-site scripting and HTML injection concepts
- Input validation, output encoding and sanitisation
- Content Security Policy concepts

SECURELAB PRACTICAL

Demonstrate vulnerabilities in authorised labs, inspect root causes, remediate and retest.

LEARNING OUTCOMES

Explain injection root causes
Demonstrate attacks safely
Verify secure remediation

EVIDENCE
Attack–remediate–retest evidence

MODULE 08 / WEEK 8

Authentication, Sessions & Access-Control Security

AIM

Understand identity, authorisation boundaries and access-control failures.

KEY CONTENT

- Authentication versus authorisation
- Password-handling concepts
- Session and cookie security
- IDOR / broken object access; RBAC

SECURELAB PRACTICAL

Test weak access-control scenarios, add server-side checks and retest.

LEARNING OUTCOMES

Identify access-control failures
Implement RBAC checks
Harden session handling

EVIDENCE
Access-control security lab

03 / MODULE GUIDE — PART 5

Web, Server & File Security

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 09 / WEEK 9

Web & Server-Side Attacks

AIM

Explore request-forgery and server-side request risks and their controls.

KEY CONTENT

- CSRF and anti-CSRF controls
- SSRF and trust boundaries
- Request manipulation and open redirects
- URL validation, allowlisting, SameSite

SECURELAB PRACTICAL

Run controlled request manipulation, apply mitigations and verify results.

LEARNING OUTCOMES

Explain CSRF/SSRF risk
Apply controls
Retest attack paths

EVIDENCE
Web attack/remediation evidence

MODULE 10 / WEEK 10

File, Configuration & Web Security

AIM

Address file handling, configuration and browser-facing weaknesses.

KEY CONTENT

- Path traversal; LFI/RFI concepts
- File uploads and validation
- Security headers and secure cookies
- Error handling, leakage and hardening

SECURELAB PRACTICAL

Analyse insecure file/configuration scenarios and document before/after.

LEARNING OUTCOMES

Recognise file/config weaknesses
Implement safer controls
Reduce information leakage

EVIDENCE
Hardening checklist + lab

03 / MODULE GUIDE — PART 6

Secure Engineering & Capstone

Module aims, core topics, SecureLab practical work, expected outcomes and portfolio evidence.

MODULE 11 / WEEK 11

Secure Programming & Secure SDLC

AIM

Bring secure coding patterns into the secure development lifecycle.

KEY CONTENT

- Secure requirements and threat modelling
- Validation, encoding and parameterised queries
- Authentication, authorisation and secrets
- Logging, exceptions and security verification

SECURELAB PRACTICAL

Transform an intentionally insecure application into a more secure implementation.

LEARNING OUTCOMES

Apply secure coding patterns
Embed security into SDLC
Document verified remediation

EVIDENCE
Secure code remediation

MODULE 12 / WEEK 12

SecureLab Attack–Defend Capstone

AIM

Demonstrate integrated discovery, exploitation, remediation, retesting and reporting.

KEY CONTENT

- Application mapping and prioritisation
- Controlled exploitation and evidence
- Root-cause analysis and secure code changes
- Retesting and professional reporting

SECURELAB PRACTICAL

Complete an authorised SecureLab Attack–Defend Challenge.

LEARNING OUTCOMES

Find/prioritise vulnerabilities
Implement and verify fixes
Communicate findings professionally

EVIDENCE
Capstone report and evidence pack

04 / SIX PRACTICE DOMAINS

One framework. Six real-world contexts.

Transfer security knowledge from controlled exercises to familiar application scenarios.

E-COMMERCE

SQL injection • XSS • authentication

APPLY • REVIEW • REMEDIATE

BANKING

Sessions • IDOR • CSRF

APPLY • REVIEW • REMEDIATE

HEALTHCARE

RBAC • privacy • access control

APPLY • REVIEW • REMEDIATE

TRAVEL & TOURS

Booking manipulation • SQLi

APPLY • REVIEW • REMEDIATE

E-LEARNING

Privilege • file uploads

APPLY • REVIEW • REMEDIATE

JOB PORTAL

XSS • uploads • information exposure

APPLY • REVIEW • REMEDIATE

CAPSTONE STANDARD

Find the weakness, explain the root cause, implement a suitable control, and demonstrate that the original attack path no longer succeeds.

05 / INTEGRATED SECURELAB METHOD

Learn the full security lifecycle

A repeatable, authorised process links offensive reasoning with defensive engineering.

01
MAP

Application & attack-surface mapping

02
IDENTIFY

Vulnerability discovery and priority

03
EXPLOIT

Controlled authorised demonstration

04
EVIDENCE

Requests, screenshots and findings

05
ANALYSE

Vulnerable code and root cause

06
REMEDIATE

Secure programming and hardening

07
RETEST

Repeat the original attack path

08
VERIFY

Record results and report impact

06 / CAPSTONE PORTFOLIO

The evidence that proves your skills

Portfolio artefacts demonstrate practical understanding, responsible conduct and verified remediation.

- Application and attack-surface map
- Vulnerability register with prioritised findings
- Screenshots or request/response evidence from authorised tests
- Root-cause notes linked to vulnerable code or configuration
- Secure implementation and remediation record
- Retest evidence showing pass/fail after remediation
- Professional technical report and concise executive summary

VERIFICATION PRINCIPLE

A security fix is not complete until the learner repeats the original attack and records evidence showing whether mitigation is effective.

07 / ASSESSMENT

Assessment & learner guidance

The original assessment weighting is preserved, with a final logbook, capstone assignment report and presentation.

SecureLab practical exercises
Guided attack, remediation and retest
30%

Technical foundations practical
Python, Flask, networking and SQL
20%

Ethical hacking practical
Authorised validation and reporting
20%

Secure programming assessment
Code-level remediation
10%

Final SecureLab capstone
Integrated Attack–Defend–Verify
20%

FINAL ASSESSMENT DELIVERABLES

01 Practical laboratory logbook
02 Capstone assignment / technical report
03 Project presentation and demonstration

08 / CAREER DIRECTION

UK job prospects & salary benchmarks

Customised from your supplied salary-card reference. These are market indicators, not guaranteed outcomes.

Junior Penetration Tester

£24k–£41k

Typical starting range

Ethical testing, vulnerability discovery and reporting.

Cybersecurity Analyst

£25k–£35k

Typical starting range

Monitoring, risk assessment and security operations.

Application Security Analyst

~ £60k

UK median advertised benchmark

Application risk, secure coding and remediation.

Security Engineer

£45k / £60k / £80k

25th / median / 75th percentile

Technical security controls across systems and cloud.

DevSecOps Engineer

£52k / £62k / £86k

25th / median / 75th percentile

Secure software delivery, CI/CD and automation.

Application Security / AppSec

£59k / £75k / £100k

25th / median / 75th percentile

Threat modelling, developer guidance and vulnerability management.

Benchmark figures reproduced from the user-supplied career reference; verify rates and source dates before publication.

09 / OUTCOMES & NEXT STEPS

A career-ready foundation

The programme is a starting point for continued study, professional practice and evidence-based career progression.

- Build Python/Flask applications and basic relational databases.
- Explain networking and web communications in application security.
- Find and demonstrate common vulnerabilities in authorised SecureLab scenarios.
- Analyse root causes and implement safer coding/configuration.
- Retest the original attack path and verify remediation.
- Produce professional findings, a logbook and a capstone report.

POTENTIAL PROGRESSION

AppSec trainee • Junior penetration-testing pathway • Vulnerability assessment • Secure development / DevSecOps foundation • Cybersecurity operations • Further specialist study.

TUITION: BRIT COLLEGE OF ENGINEERING & TECHNOLOGY

https://bcet.uk

R&D / CAPSTONE: CARSIT UK | https://carsit.bcet.uk

ENDORSEMENT: ISM EDUCATION | https://ismeducation.com