Instructor-led Training
Course Catalog
January 2019

800.873.8193
sig-info@synopsys.com
synopsys.com/software
The Synopsys Difference

Synopsys offers the most comprehensive solution for building integrity—security and quality—into your SDLC and supply chain. We've united leading testing technologies, automated analysis, and experts to create a robust portfolio of products and services. This portfolio enables companies to develop customized programs for detecting and remediating defects and vulnerabilities early in the development process, minimizing risk and maximizing productivity. Synopsys, a recognized leader in application security testing, is uniquely positioned to adapt and apply best practices to new technologies and trends such as IoT, DevOps, CI/CD, and the Cloud. We don't stop when the test is over. We offer onboarding and deployment assistance, targeted remediation guidance, and a variety of training solutions that empower you to optimize your investment. Whether you're just starting your journey or well on your way, our platform will help ensure the integrity of the applications that power your business.

For more information go to www.synopsys.com/software.

Synopsys, Inc.
185 Berry Street, Suite 6500
San Francisco, CA 94107 USA

U.S. Sales: 800.873.8193
International Sales: +1 415.321.5237
Email: sig-info@synopsys.com
Introduction

Our Curriculum

Suggested Course Sequencing

Introductory

Attack and Defense

OWASP Top 10

Principles of Software Security

Defending

Defending Android

Defending C#.NET Core

Defending C#.ASP.NET

Defending C# Non-web

Defending C or C++

Defending Golang

Defending HTML5

Defending iOS

Defending JavaEE

Defending JavaSE

Defending JavaScript

Defending PHP

Defending Python

Securing Mobile Platforms

Principles of Software Security for COBOL

Securing

Securing Mobile Platforms 101

Securing Mobile Platforms 201

Securing AWS

Securing Azure

Securing Containers

Securing OAuth 2.0 and OpenID/Connect

Securing Web Services

Attacking

Attacking Code Using Static Analysis

Attacking Networks

Attacking Web Applications

Workshops

Embedded Security

Red Teaming

Threat Modeling
INTRODUCTION

Synopsys’ instructor-led courses are developed and taught by experts at the forefront of the software security field. Our instructors are certified security professionals who have hands-on experience working directly with clients on their security challenges.

Our curriculum includes training modules for professionals just starting out with software security as well as those who are looking to develop more advanced skills. Synopsys continuously develops its courses to accommodate the rapid changes in software security.

OUR CURRICULUM

Synopsys’ curriculum is a series of complementary courses designed to meet your organization’s needs. You can select the courses that best match your level of experience, your role, and the development platforms in your organization. Our courses are grouped into the following software security activities:

1. **Introductory** – New to software security? Consider our Principles of Software Security or Attack and Defense course.

2. **Defending** – Learn defensive programming skills in context, in specific languages targeted to specific development platforms, so you can defend against attacks in your code.

3. **Securing** – Learn how to identify common vulnerabilities and essential strategies to secure your deployment and tech stack from external threats.

4. **Attacking** – Use your knowledge to test your applications for security vulnerabilities.


We can work with you to select a curriculum that is right for your organization.
Synopsys' software security curriculum provides valuable knowledge across every role within software development organizations. Our instructor-led training features a broad library of over 30 courses, so you can design a long term plan to increase the security knowledge and skills of everyone within your SDLC.

Below you’ll find some sample learning paths for developers, engineers, and architects. Pick and choose the courses your developers need, or design your own learning path—it’s up to you.

<table>
<thead>
<tr>
<th>Application Developers</th>
<th>DevSecOps Engineers</th>
<th>Mobile Developers</th>
<th>QA Engineers</th>
<th>Architects</th>
<th>Security Practitioner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defending JavaScript Defending HTML5 Defending GoLang</td>
<td>Defending Python Defending PHP Defending Java Defending GoLang</td>
<td>Defending iOS Defending Android</td>
<td>OWASP Top 10</td>
<td>Securing OAuth 2.0 &amp; OpenID/Connect</td>
<td>Attacking Web Applications Attacking Code w/ Static Analysis Attacking Networks</td>
</tr>
<tr>
<td>Principles of Software Security Cobol</td>
<td>Attacking Code w/ Static Analysis</td>
<td>Attacking Code w/ Static Analysis</td>
<td>Attacking Code w/ Static Analysis</td>
<td>Red Teaming Workshop</td>
<td></td>
</tr>
<tr>
<td>Attacking Code w/ Static Analysis</td>
<td>Threat Modeling</td>
<td></td>
<td></td>
<td></td>
<td>Threat Modeling</td>
</tr>
</tbody>
</table>

---

*Introductory* *Intermediate* *Advanced*
IN-PERSON OR VIRTUAL — YOUR CHOICE

If you choose traditional instructor-led training, our certified instructors will travel to the location of your choice. Our instructors are trained to engage your audience through group discussion and interactive hands-on labs designed to simulate real-world environments. On-site instructors can make course adjustments to better complement the needs, interests, and experience-level of your participants.

If you have a distributed workforce, your participants can avoid travel and time away from the office using our Virtual Instructor-led Training (VILT). VILT is separated into shorter sessions to optimize participant engagement. VILT can be delivered over consecutive working days or on a weekly basis, depending on your team’s preference. Virtual training is a cost-conscious alternative for supporting your employees’ professional development.

Instructor-led courses are held on your schedule in the format that works best for you.

Synopsys utilizes a number of training strategies to assist in participant engagement, including hands-on labs using our cloud-based VM solution, breakout groups, live demonstrations, white boarding, videos, polling, and real-time assessments.
Attack and Defense

Description

This course provides software builders and testers an in-depth look at standard attacks and their corresponding defenses. It empowers students of this course to solve tricky problems securely in their own environment by mapping them to known problems and tried and tested solutions.

This course introduces common attacks that most applications deal with. These attacks are cast into different contexts such as web, embedded, thick client, or mobile, and their standard solutions are discussed in the classroom. Students are then guided to apply this knowledge to identify attacks and design defenses for a model application throughout the labs.

Intended Audience

» Developers
» Architects
» QA Engineers
» Security Practitioners

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours
Description

This course focuses on the most important security defects found in web applications, covering all issues in the latest OWASP Top 10 list. Each topic describes a vulnerability and provides practical guidance for remediation. This course also provides demonstrations and practical hands-on exercises where students learn what impact these security issues can have on web applications.

Intended Audience

» Developers
» Architects
» QA Engineers
» Security Practitioners

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours
Principles of Software Security

Description

The key to proactive computer security involves getting a risk management handle on the software security problem. This half-day course, created by the experts who literally wrote the book on software security, encompasses software security awareness and best practices for a general audience. Everyone involved in software production requires baseline knowledge of software security problems and risks, along with an overall understanding of approaches for producing better software.

This virtual classroom course is available in two flavors:

The Vanilla flavor explains common problems in software security and describes an approach to infusing software security into the development process through risk management, software security touchpoints, and historical knowledge of software security vulnerabilities.

The Requirements flavor focuses on introducing important cost-saving software security requirements early in the software development life cycle.

Intended Audience

» Developers
» Architects
» QA Engineers
» Security Practitioners

Delivery Format

» Virtual Classroom

Class Duration

» 4 hours
Description

The first part of this course begins with foundational knowledge about the Android platform, its architecture, and the security model. Then it builds on that knowledge to discuss Android-specific risks. The next part delves into defensive programming techniques for application risks specific to the Android platform and common application security risks. Also discussed here is the risk itself, code which implements it, and time-tested techniques for addressing these risks.

Intended Audience
- Mobile Developers
- DevSecOps
- Architects

Delivery Format
- Traditional Classroom
- Virtual Classroom

Class Duration
- 8 hours

Description

This course is focused on modern .NET core secure development with an emphasis on microservices, service-oriented architecture and cloud-first applications. You will learn modern attacker techniques and how to defensively write code to prevent these vulnerabilities in your applications. This course also discusses different activities that can be performed during the secure development lifecycle (SDLC) that help detect and prevent the vulnerabilities.

Intended Audience
- Developers
- DevSecOps
- Architects

Delivery Format
- Traditional Classroom
- Virtual Classroom

Class Duration
- 8 hours
Description
This course explores in detail the root cause and remediation of vulnerabilities that apply to ASP.NET C# web applications. It also discusses different activities that can be performed during the secure development lifecycle (SDLC) that help detect and prevent the vulnerabilities such as penetration testing and code review.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Traditional Classroom
» Virtual Classroom

Class Duration
» 8 hours

Description
The Defending C# for Desktop Applications course explores in detail the root causes and remediation of vulnerabilities applicable to ASP.NET C# non-web applications. Students learn about inherent risks when using .NET, including how to process user-input securely, the security risks related to determinism and concurrency, the risks associated with file access and inter-process communication, UI programming, assembly loading, and safe error handling. The purpose of the .NET application security lab is to learn best practices and write secure code. It covers XML and threading problems.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Traditional Classroom
» Virtual Classroom

Class Duration
» 8 hours
Defending C or C++

Description

This course provides developers with a strong foundation in software security as it relates to the implementation of applications developed in C or C++. It includes detailed examples and focuses on the correct way to think through security problems by combining structured theory, demonstrations, technical deep-dives, and illustrated explanations. This course connects the habit of building security in through proven programming practices and explains common security-related problems in detail so that students can avoid them in their own work. The course can be delivered either as a C flavor or C++ flavor, but not both.

Intended Audience

» Developers
» DevSecOps
» Architects

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours or 16 hours

Defending Golang

Description

Defending Golang is a full-day course that addresses the questions on the secure development of systems software and web applications in Golang. It helps attendees understand generic risks in highly performant, concurrent systems software, as well as specific risks for Golang developers and the specific remedies available to them. This course also explains risks present in developing web applications and web services in Golang. Using real vulnerabilities from production open source projects, this course demonstrates the real-world complexity of security risks and remediation.

Intended Audience

» Developers
» DevSecOps
» Architects

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours
Defending HTML5

**Description**

As HTML5 gains relevance in enterprises for its rich features and enhanced online experience, developers need to consider the vulnerabilities that their new code may introduce into their organizations' web infrastructure.

The Defending HTML5 course addresses this concern by identifying risks in the new HTML5 landscape and explaining how to mitigate them with defensive programming strategies and best practices.

**Intended Audience**
- Developers
- DevSecOps
- Architects

**Delivery Format**
- Traditional Classroom
- Virtual Classroom

**Class Duration**
- 8 hours

Defending iOS

**Description**

The course begins with an overview of the iOS platform, the securities that are built in, and how they have evolved over the many iterations of the iOS operating system. Next, the course walks through common iOS application security concerns and discusses how best to mitigate or remediate such issues. The course looks at the risk, the code that implements the risk, as well as code examples for the issues and remediation steps.

NOTE: Customers are requested to pre-select the programming language relevant to them—whether Objective-C or Swift—when ordering this course.

**Intended Audience**
- Mobile Developers
- DevSecOps
- Architects

**Delivery Format**
- Traditional Classroom

**Class Duration**
- 8 hours
Defending Java EE

Description

This course focuses on using defensive programming techniques in Java EE applications against common web vulnerabilities. It discusses an approach to identify security risks and vulnerabilities, apply defensive programming techniques, and securely configure web applications.

This course also provides demonstrations and practical hands-on exercises where students learn how to identify security vulnerabilities in the code and fix them using best practices discussed in the course.

Intended Audience

» Developers
» DevSecOps
» Architects

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours
Defending JavaScript

Description
Defending JavaScript is a full-day course that addresses the questions of secure development in front-end and back-end JavaScript. It helps attendees understand generic web application risks as well as specific risks involved in manipulating JavaScript in the DOM (Document Object Model) on the client side, bypassing browser controls such as same origin policy and sandboxing, sending Ajax requests, and using client-side frameworks and libraries. This course also explains risks present in server-side code written in JavaScript, and covers security best practices, different types of injections, framework misconfigurations, cross-site request forgery, insecure deserialization, output encoding, and input validation.

NOTE: Customers are requested to pre-select the client-side framework relevant to their development environment—whether Angular, AngularJS, or React—when ordering this course.

Node.js and Express.js server-side frameworks are available with any client-side framework selection.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Traditional Classroom
» Virtual Classroom

Class Duration
» 8 hours

Defending PHP

Description
Defending PHP is a full-day course that covers the challenges of secure development in PHP for web applications. The course examines the risks associated with the misconfiguration of the PHP instance, the misuse of PHP functions and APIs that are not secure by default, solutions for common types of web application vulnerabilities in the PHP context, and broader secure architecture topics, such as securing the SDLC and secure design.

Upon completion, students will be able to appreciate security risks common to PHP applications, identify security vulnerabilities in PHP environments and applications, and write secure PHP code.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Traditional Classroom
» Virtual Classroom

Class Duration
» 8 hours
Defending Python

Description
Python is an extensively used general-purpose, high-level programming language that is designed to support rapid development, prototyping, and fast iterative development. The course focuses on general Python security topics, but also considers areas such as Django or Flask for those working with Python web technologies. The goal of this training is to focus on security attack vectors, application protections, security principles and best practices that can be used while developing Python applications. It is divided into multiple lecture modules with corresponding lab exercises.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Traditional Classroom
» Virtual Classroom

Class Duration
» 8 hours

Principles of Software Security for COBOL

Description
Software in any development and execution environment is subject to intrusion. This is certainly true of the Common Business Oriented Language (COBOL) mainframe environment. The course has two main sections. The first section, “Foundations of COBOL Security,” looks at security risks to your company and how your code might contribute to that risk. The second section, “COBOL Defensive Programming Techniques,” is the core of this course. It looks at COBOL programming best practices and then talks about how to avoid or mitigate the vulnerabilities described in the taxonomy of vulnerabilities. This includes examples of bad code snippets and mitigation for each vulnerability with secure coding examples to avoid the vulnerability.

Intended Audience
» Developers
» DevSecOps
» Architects

Delivery Format
» Virtual Classroom

Class Duration
» 4 hours
Securing Mobile Platforms 101

Description

This course contains three modules:

- Mobile Trends: Mobile First AppSec
- Overview of Mobile Platforms
- Secure Development

It is aimed at users who are new to mobile, new to mobile security, or have had brief exposure to such topics. We cover areas such as common mobile vulnerabilities and mitigating controls, different app architecture types, and more.

Intended Audience

- Mobile Developers
- Architects
- QA Engineers
- Security Practitioners

Delivery Format

- Traditional Classroom
- Virtual Classroom

Class Duration

- 8 hours
Description

This course contains three modules:

- Mobile Payments
- Protecting Client-Side Code
- Protecting Business Critical Applications

The course is aimed at users with some understanding of mobile platforms who want to gain an understanding of advanced security concepts. This course places particular emphasis on applications that are highly sensitive or are considered business critical. During this course, we delve into tampering of mobile apps, jailbreaking, rooting, the use of debuggers, and more.
Securing AWS

Description

Cloud computing has grabbed the world’s attention not only for its pervasive, on-demand, convenient usage, but for its ability to be vulnerable to data breaches and novel forms of attack. Since most software uses the cloud in various shared capacities (development, hosting, or integration with third-party code), threats from hackers are inevitable. This hands-on workshop equips students to understand this new landscape of converged infrastructure and shared services, its existing and emerging threats, and provides them with secure mitigation methods.

The Securing AWS course is an introductory course, covering Amazon core services, such as IAM, KMS, EC2, S3 and VPCs, with a focus on security. The course enables students to identify areas for cross-pollination between development and operations that enhance application, infrastructure, and network security.
Securing Azure

Description

In this course, you learn how to secure your Azure Infrastructure as a Service (IaaS). This course initially presents a brief overview of the different Azure infrastructure offerings, and then dives into how to secure them. In hands-on labs, students learn how to spot an insecure Azure configuration and subsequently fix it. The course covers the following topics:

- Identity and Access Management
- Storage Security
- Network Security
- Compute Security

This course is aimed at users of all levels of familiarity with Azure, and does not require any prior Azure knowledge.

NOTE: This course is not intended to teach cloud or Azure concepts and offerings. It does, however, provide a brief introduction to several important infrastructure components in Azure, and focuses on how to secure them.

Intended Audience

- Developers
- DevSecOps
- Architects
- Security Practitioners

Delivery Format

- Traditional Classroom
- Virtual Classroom

Class Duration

- 8 hours
Description

Containers have changed the way applications are being deployed nowadays. In the past few years, the reason the term "Containerization" has gained a lot of traction is because it makes the job easy for an application team to build, package and distribute a microservice or an application across different environments. Docker has emerged as the leading container technology that is used by organizations large and small for packaging and deploying these services or applications.

However, as always, security is a challenge that organizations face when it comes to deploying containers securely. Container security refers to the protection of the integrity of the containers which include both the application as well as the infrastructure they make use of. This course features hands-on labs and instruction that will enable students to harden the container runtime and the container host, along with a host of other best practices pertaining to container security.
Securing OAuth 2.0 and Open ID/Connect

Description

In modern web applications, OAuth 2.0 enables users to give applications access to (a subset of) their data hosted on a website without sharing their passwords. OpenID/Connect is built on top of OAuth 2.0 and enables a user to authenticate to the application using the credentials of an identity provider. However, creating or interacting with an OAuth 2.0 or an OpenID/Connect-enabled server is not without its security pitfalls. In this course, you learn how to avoid the most common defects when creating or interacting with an OAuth 2.0/OpenID/Connect server.

Intended Audience

» Developers  
» DevSecOps  
» Architects  
» Security Practitioners

Delivery Format

» Traditional Classroom  
» Virtual Classroom

Class Duration

» 8 hours
Securing Web Services

Description

The Securing Web Services course is intended for developers, engineers, and architects who work with backend web services APIs which may not necessarily have a User Interface (UI) or a UI component.

This course examines web services concepts and then takes a deep dive into several web services technologies such as WS-Security, Security Assertion Markup Language (SAML), and OAuth. This course also covers risks inherent to web services and how to properly threat model web services. Web service security is examined from the perspective of the message, the channel, and the service itself.

The lab component of this course allows students to gain an understanding of and practice with some of the real-world security issues inherent to web services. The lab is intentionally written with a programming language and framework that are popular but with which most developers are not familiar: Python and Flask. This allows students to focus on secure design and secure coding concepts without being too concerned with the implementation details of a particular language.

This is a comprehensive and stand-alone course on web services. Many concepts taught in this course are covered in depth in other courses, such as Threat Modeling and OAuth. If you are building a multi-day curriculum for web service developers, please reach out to the Synopsys training team for advice on course selection.
Attacking Code Using Static Analysis

Description

This course focuses on the static analysis process and tools that can be used to test and attack a web application. It explains the static analysis techniques, compares manual and automated code reviews, and discusses the implementation of static analysis in your SDLC.

The course also provides demonstrations and practical hands-on exercises where students learn how to identify common vulnerabilities using code review and common static analysis tools.

Intended Audience

» Developers
» Architects
» Security Practitioners
» QA Engineers

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours
Attacking Networks

Description

The Attacking Networks course explains how to test for security issues on a network. It covers introductory network concepts, defines network security testing methodology, and describes what the testing process looks like, including the common techniques used during network penetration tests.
Description

The Attacking Web Applications course explains how to test for security issues in web applications. It defines what web security testing is and how it differs from other forms of testing, describes what the testing process looks like, and gives specific guidance on how to test for some of the most important risks in web applications.
Embedded Security Workshop

Description

This course provides an introduction into security engineering for professionals that develop embedded systems, IoT (Internet of Things), or other integrated systems. The course is geared towards those students who have a firm understanding of the principles of designing, engineering, or developing non-IT systems and seek to understand the influence of security as a stakeholder in design.

Students are provided with a base understanding of cyber-security as it relates to various systems and the processes that should be present within their engineering lifecycles. The course takes the approach of understanding risks and vulnerabilities typically present in these systems, and outlining processes and techniques to assist in developing software and embedded systems to minimize cyber security risk.

Intended Audience

» Developers
» Architects
» Security Practitioners

Delivery Format

» Traditional Classroom

Class Duration

» Varies
Red Teaming Workshop

Description

Red Teaming is a goal-based assessment approach which allows organizations to gain insight into how their security posture is when faced with a real threat. This hands-on, two day course introduces students to the concepts of Red Teaming and how it is different from traditional vulnerability testing. The course also includes guidance for the organization on creating and maintaining its own internal Red Teams. Students in this course are introduced to physical, social, and electronic testing methods which can be utilized during Red Team engagements.

Intended Audience

» Developers
» Architects
» Security Practitioners

Delivery Format

» Traditional Classroom

Class Duration

» 8 or 16 hours
Threat Modeling Workshop

Description

Discovering weaknesses in the design of a system is the specific goal of threat modeling. Synopsys’ threat modeling approach can reveal security issues not fully addressed by the traditional methods of penetration testing and secure code review. Organizations benefit from this software design analysis because you can perform it without code to discover potential vulnerabilities early in the development cycle.

Intended Audience

» Developers
» DevSecOps
» Architects
» QA Engineers
» Security Practitioners

Delivery Format

» Traditional Classroom
» Virtual Classroom

Class Duration

» 8 hours