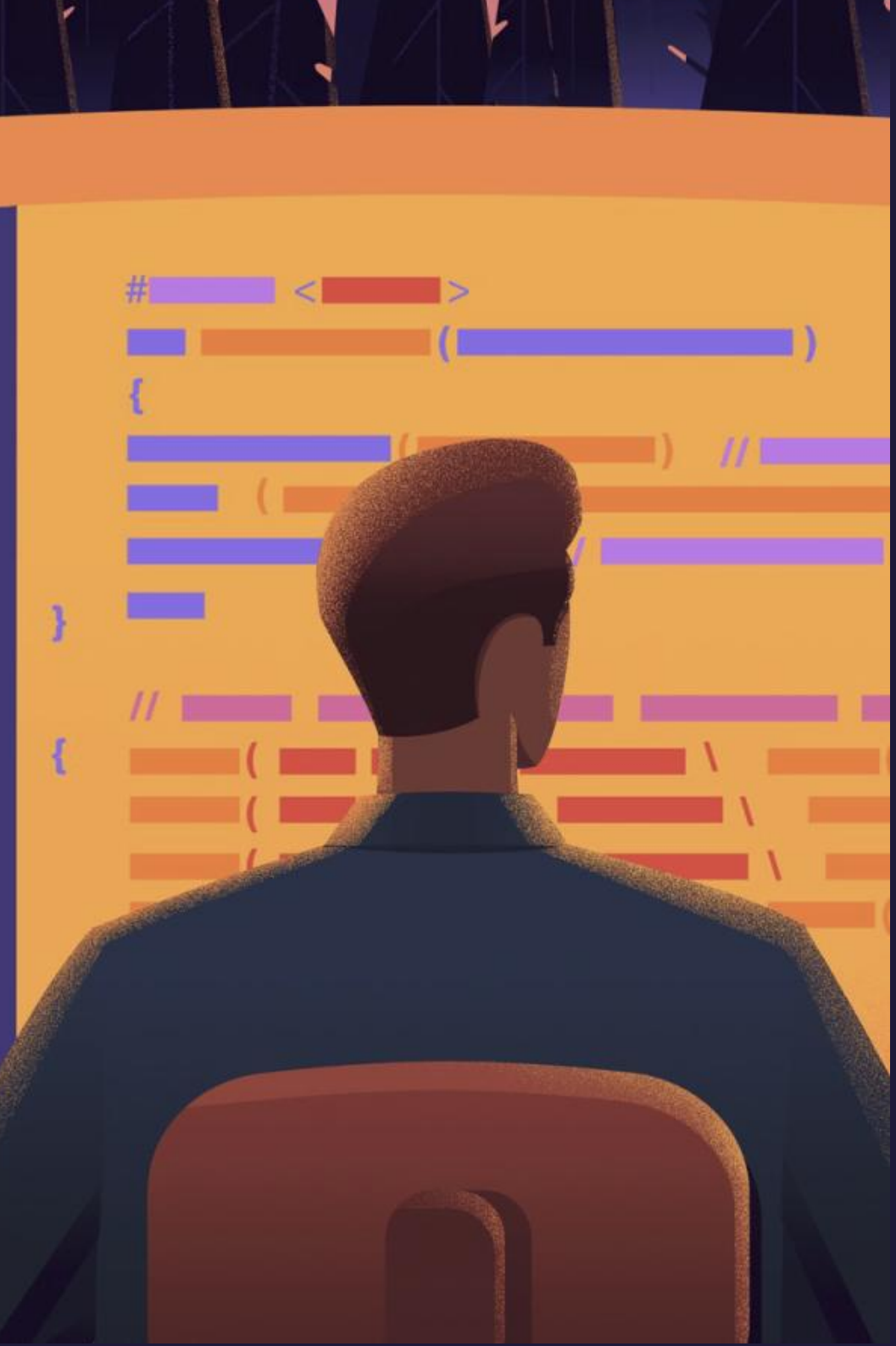

بررسی ساختار دامنه هشتم CISSP

ذهنیت سازی امنیت توسعه نرم‌افزار

دامنه هشتم

Software Development Security

8

دامنه 8 از آزمون CISSP امنیت توسعه نرم‌افزار (Software Development Security)

دامنه هشتم از گواهینامه CISSP به امنیت توسعه نرم‌افزار (Software Development Security) می‌پردازد که شامل درک دقیق چرخه عمر سیستم، فرآیند توسعه نرم‌افزار، مدیریت آسیب‌پذیری‌ها و پیاده‌سازی روش‌های امن توسعه است. این حوزه هم‌زمان با پیشرفت سریع DevOps و توسعه سریع نرم‌افزارها، یکی از مهم‌ترین بخش‌های امنیت سایبری به‌شمار می‌رود.

بخش اول SDLC &System Life Cycle (SLC)

چرخه عمر سیستم (SLC) و چرخه حیات توسعه نرم‌افزار (SDLC) چهارچوب‌هایی هستند که برای تولید و نگهداری نرم‌افزارها به‌کار می‌روند. مراحل SDLC عبارت‌اند از:

- **برنامه‌ریزی (Planning)** تعریف اهداف امنیتی و نیازمندی‌ها
- **تحلیل نیازمندی‌ها (Requirements)** استخراج نیازهای امنیتی و عملیاتی
- **طراحی معماری (Architecture & Design)** طراحی امن سیستم، مدل‌سازی تهدیدات
- **پیاده‌سازی (Development)** استفاده از اصول برنامه‌نویسی امن، اعتبارسنجی ورودی‌ها
- **تست و ارزیابی (Testing & Evaluation)** استفاده از تست‌های نفوذ، تست واحد، تست یکپارچه و تایید امنیتی
- **استقرار (Deployment)** تایید نهایی، صدور مجوز استفاده، بررسی سیاست‌های امنیتی
- **عملیات و نگهداری (Operations & Maintenance)** بررسی لاگ‌ها، به‌روزرسانی‌ها، اصلاح آسیب‌پذیری‌ها
- **نابودی (Disposal)** حذف امن داده‌ها و اجزای نرم‌افزار

بخش دوم مدل‌های توسعه نرم‌افزار

- **Waterfall**مدل سنتی، امکان برگشت به مراحل قبلی نیست.
- **Agile**توسعه مبتنی بر چرخه‌های کوتاه(Sprint) ، با تمرکز بر انعطاف‌پذیری.
- **Scrum**چارچوبی در Agile با نقش‌هایی مانندScrum Master
- **DevOps**یکپارچه‌سازی توسعه، عملیات و تضمین کیفیت.
- **SecDevOps** افزودن امنیت به چرخهDevOps
- **Canary Testing** پیاده‌سازی آزمایشی تغییرات در بخشی از سیستم.

بخش سوم آسیب‌پذیری‌ها و ضعف‌های رایج نرم‌افزاری

- Buffer Overflow
- SQL Injection
- Cross-Site Scripting (XSS) / Cross-Site Request Forgery (CSRF)
- Backdoor / Trapdoor
- Memory Reuse / Object Reuse
- Time-of-check to time-of-use (TOCTOU)
- Covert Channels
- Obfuscation پنهان‌سازی کد
- Session Management ناامن

بخش چهارم امنیت در کدنویسی و طراحی

- **Input Validation**جلوگیری از ورود داده‌های غیرمجاز
- **Secure Session Management**مدیریت توکن‌ها، تایم‌اوت‌ها و اعتبارسنجی
- **Polyinstantiation**جلوگیری از افشای اطلاعات در سطوح مختلف دسترسی
- **Software Configuration Management (SCM)**مدیریت نسخه و تغییرات کد
- **Citizen Developers**کاربران غیرمتخصصی که با ابزارهای کم‌کدنویسی نرم‌افزار می‌سازند؛ بررسی ریسک‌ها الزامی است.

بخش پنجم امنیت پایگاه داده‌ها

- **ACID (Atomicity, Consistency, Isolation, Durability)** اصول کلیدی تراکنش‌های امن
- **Primary Key / Foreign Key** طراحی مناسب جداول و روابط برای جلوگیری از ناسازگاری داده‌ها
- **Locks & Concurrency** کنترل همزمانی برای جلوگیری از شرایط رقابتی (Race Conditions)
- **Injection Attacks:** محافظت در برابر SQL Injection از طریق اعتبارسنجی و استفاده از ORM ها

بخش ششم امنیت API و وب‌سرویس‌ها

- **RESTful APIs** استفاده از روش‌های HTTP با فرمت JSON/XML
- **SOAP** پروتکل قدیمی‌تر مبتنی بر XML
- **امنیت در API** اعتبارسنجی توکن‌ها، محدودسازی نرخ درخواست، بررسی دسترسی‌ها

بخش هفتم مدل‌های بلوغ و تضمین کیفیت

- **Maturity Models:** مانند CMMI برای سنجش بلوغ فرآیندهای توسعه نرم‌افزار
- **Certification / Accreditation:** تاییدیه‌های امنیتی برای نرم‌افزارها

توصیه‌های امنیتی و منابع یادگیری

- مطالعه راهنمای OWASP برای توسعه‌دهندگان
- استفاده از ابزارهای بررسی کد مانند SonarQube یا Fortify
- تحلیل آسیب‌پذیری با استفاده از SAST و DAST
- پیاده‌سازی SIEM برای لاگ‌گیری از کدهای در حال اجرا