المؤسسة العربية للاستشارات والتدريب وتطوير الأداء
AINFCT I info@ainfct.com I www.ainfct.com

Secure Coding Practices for Developers

فكرة الدورة

أصبحت البرمجيات محوراً أساسياً في الخدمات الرقمية والأنظمة المؤسسية، ومع تزايد الاعتماد على التطبيقات وواجهات البرمجة والخدمات السحابية، ازدادت الحاجة إلى دمج الأمن داخل عملية التطوير من مراحل التصميم الأولى حتى الاختبار والنشر والصيانة. لم تعد البرمجة الآمنة نشاطاً منفصلاً عن التطوير، بل أصبحت مسؤولية عملية ترتبط بجودة الكود، وإدارة الثغرات، وحماية البيانات، وتقليل مخاطر الاختراق.

يركز هذا البرنامج التدريبي من AINFCT على تمكين المطورين من فهم ممارسات البرمجة الآمنة وتطبيقها داخل دورة حياة تطوير البرمجيات. ويتناول البرنامج مبادئ التصميم الآمن، والتحقق من المدخلات، وإدارة الهوية والجلسات، والتحكم في الوصول، والتشفير، وأمن واجهات البرمجة، وأمن قواعد البيانات، والاختبار الأمني للكود.

يوفر البرنامج قيمة عملية من خلال ربط الممارسات الأمنية اليومية بسيناريوهات تطوير واقعية تساعد الفرق على إنتاج برمجيات أكثر موثوقية وقابلية للصيانة.

أهداف الدورة

فيما يلي الأهداف الرئيسية لهذا البرنامج التدريبي:

- تطبيق مبادئ البرمجة الآمنة داخل دورة التطوير.
- تحليل مخاطر الكود والثغرات التطبيقية الشائعة.
- استخدام ضوابط فعالة للتحقق من المدخلات.
- تأمين المصادقة والجلسات والتحكم في الوصول.
- تحسين أمان واجهات البرمجة وقواعد البيانات.
- دمج الاختبار الأمني في ممارسات التطوير.

منهجية الدورة

- شرح تفاعلي يربط المفاهيم الأمنية بسياقات التطوير اليومية.
- تمارين مراجعة كود لاكتشاف الأنماط البرمجية غير الآمنة.
- تحليل حالات عملية لثغرات تطبيقية شائعة.
- أنشطة جماعية لبناء قوائم تحقق أمنية قابلة للاستخدام.
- مناقشات موجهة حول دمج الأمن في مسارات DevSecOps.

أثر الدورة على المنظمة

يمكن رفع جودة البرمجيات وتقليل المخاطر التشغيلية من خلال:

- تقليل الثغرات الناتجة عن أخطاء التطوير.
- تحسين موثوقية التطبيقات والخدمات الرقمية.
- تعزيز التكامل بين التطوير والأمن والجودة.
- دعم الامتثال لمتطلبات أمن التطبيقات.

أثر الدورة على المتدربين

يساعد البرنامج المشاركين على:

- كتابة كود أكثر أماناً وقابلية للمراجعة.
- فهم أنماط الهجمات على التطبيقات.
- اختيار ضوابط حماية مناسبة أثناء التطوير.
- التعامل المهني مع نتائج الفحص الأمني.

الشهادات

شهادة معتمدة من AINFCT

الفئة المستهدفة

يناسب هذا البرنامج المطورين وفرق البرمجيات الذين يحتاجون إلى دمج الأمن داخل ممارسات التطوير اليومية. كما يفيد الفرق الفنية التي تتعامل مع التطبيقات، وواجهات البرمجة، والكود المصدري، ونتائج الفحص الأمني.

- مطورو تطبيقات الويب والواجهات الخلفية.
- مطورو واجهات البرمجة والخدمات المصغرة.
- مهندسو البرمجيات وقادة الفرق التقنية.
- مختصو ضمان الجودة واختبار التطبيقات.
- فرق DevOps وDevSecOps الفنية.

موضوعات الدورة

اليوم الأول: مدخل إلى البرمجة الآمنة ودورة التطوير

- مفهوم البرمجة الآمنة ودورها في جودة البرمجيات.
- العلاقة بين الأمن ودورة حياة تطوير البرمجيات.
- مصادر الثغرات الناتجة عن التصميم والتنفيذ.
- مبادئ تقليل سطح الهجوم داخل التطبيق.
- مسؤوليات المطور داخل بيئة DevSecOps.

اليوم الثاني: التفكير الأمني ونمذجة التهديدات

- تحليل الأصول ونقاط الثقة داخل التطبيق.
- تحديد التهديدات المرتبطة بالمستخدمين والبيانات.
- استخدام سيناريوهات إساءة الاستخدام أثناء التصميم.
- تحديد الضوابط المناسبة قبل كتابة الكود.
- توثيق الافتراضات الأمنية وحدود النظام.

اليوم الثالث: التحقق من المدخلات ومعالجة البيانات

- مبادئ التحقق الإيجابي من المدخلات.
- التمييز بين التحقق والتنظيف والترميز.
- التعامل مع البيانات غير الموثوقة من المستخدم.
- منع حقن الأوامر والاستعلامات غير الآمنة.
- تصميم قواعد تحقق قابلة للصيانة.

اليوم الرابع: الترميز الآمن للمخرجات ومنع XSS

- فهم هجمات Cross-Site Scripting وأنواعها.
- اختيار الترميز المناسب حسب سياق المخرجات.
- تقليل الاعتماد على معالجة النصوص العشوائية.

- حماية القوالب والواجهات الديناميكية.
- مراجعة أمثلة عملية لأخطاء الترميز الشائعة.

اليوم الخامس: المصادقة وإدارة كلمات المرور

- تصميم تدفقات مصادقة آمنة وقابلة للاستخدام.
- تخزين كلمات المرور باستخدام خوارزميات مناسبة.
- تطبيق المصادقة متعددة العوامل عند الحاجة.
- تقليل مخاطر إعادة تعيين كلمات المرور.
- منع تعداد المستخدمين ورسائل الخطأ الكاشفة.

اليوم السادس: إدارة الجلسات والرموز الأمنية

- مبادئ إنشاء الجلسات وإبطالها بأمان.
- حماية ملفات تعريف الارتباط والرموز الحساسة.
- تقليل مخاطر سرقة الجلسات وتثبيتها.
- إدارة انتهاء الجلسة وتجديد الرموز.
- التعامل الآمن مع JSON Web Tokens.

اليوم السابع: التحكم في الوصول والصلاحيات

- التمييز بين المصادقة والتفويض.
- تطبيق مبدأ أقل امتياز داخل التطبيق.
- منع التحكم غير الآمن في الكائنات.
- إدارة الأدوار والصلاحيات بطريقة مركزية.
- اختبار قرارات الوصول في المسارات الحرجة.

اليوم الثامن: التشفير وحماية البيانات الحساسة

- اختيار الاستخدامات المناسبة للتشفير والتجزئة.
- تجنب بناء خوارزميات تشفير مخصصة.
- إدارة المفاتيح والأسرار داخل بيئة التطوير.

- حماية البيانات أثناء النقل والتخزين.
- تقليل كشف البيانات في السجلات والرسائل.

اليوم التاسع: أمن قواعد البيانات والاستعلامات

- استخدام الاستعلامات المعلمة لمنع SQL Injection.
- تقييد صلاحيات حسابات قواعد البيانات.
- حماية الإجراءات المخزنة وطبقات الوصول للبيانات.
- إدارة الأخطاء دون كشف تفاصيل حساسة.
- مراجعة أنماط شائعة في استعلامات غير آمنة.

اليوم العاشر: أمن واجهات البرمجة API

- تصميم نقاط نهاية آمنة ومحددة المسؤولية.
- حماية واجهات REST وGraphQL من الإساءة.
- تطبيق حدود المعدل والتحقق من الصلاحيات.
- إدارة المفاتيح والرموز في تكاملات الخدمات.
- توثيق المتطلبات الأمنية لواجهات البرمجة.

اليوم الحادي عشر: إدارة الأخطاء والتسجيل الآمن

- تصميم رسائل خطأ لا تكشف معلومات داخلية.
- تحديد الأحداث الأمنية الواجب تسجيلها.
- حماية السجلات من التلاعب والتسريب.
- ربط السجلات بعمليات الرصد والاستجابة.
- الموازنة بين التشخيص والخصوصية.

اليوم الثاني عشر: أمن الملفات والتحميلات والمحتوى

- التحقق من أنواع الملفات وحجمها ومصدرها.
- منع تنفيذ الملفات المرفوعة داخل الخادم.
- إدارة التخزين المؤقت للمحتوى الحساس.

- حماية مسارات الملفات من التلاعب.
- تطبيق فحص أمني للملفات عند الحاجة.

اليوم الثالث عشر: مراجعة الكود والتحليل الأمني

- مبادئ مراجعة الكود من منظور أمني.
- استخدام التحليل الساكن لاكتشاف الأنماط الخطرة.
- استخدام التحليل الديناميكي ضمن بيئات الاختبار.
- ترتيب نتائج الفحص حسب المخاطر.
- تحويل النتائج إلى تحسينات قابلة للتنفيذ.

اليوم الرابع عشر: أمن الاعتماديات وسلسلة التوريد

- إدارة المكتبات والحزم مفتوحة المصدر.
- فحص الثغرات في الاعتماديات الخارجية.
- تثبيت الإصدارات وتوثيق المكونات البرمجية.
- حماية أسرار البناء والنشر الآلي.
- تقليل مخاطر سلاسل التوريد البرمجية.

اليوم الخامس عشر: التكامل التطبيقي وخطة التحسين

- دمج الممارسات الآمنة داخل مهام التطوير اليومية.
- بناء قائمة تحقق أمنية للفريق.
- مراجعة سيناريو تطبيق متكامل متعدد الطبقات.
- تحليل أخطاء شائعة ووضع إجراءات تصحيحية.
- إعداد خطة تطوير مهني بعد البرنامج.

course_daily_schedule

يمتد البرنامج لمدة 15 يوماً تدريبياً، بواقع 4 ساعات تدريبية يومياً. يبدأ كل يوم بمراجعة مركزة للمفاهيم السابقة، ثم عرض للمحور الرئيسي، يتبعه تطبيق عملي أو تحليل كود أو مناقشة حالة، وينتهي اليوم بخلاصة عملية تربط الموضوع بسياق التطوير المؤسسي. يتم توزيع الوقت بما يوازن بين الشرح، والتطبيق، والمراجعة، وتبادل الخبرات.

course_assessment

يعتمد التقييم على المشاركة الفعالة، وتمارين مراجعة الكود، وتحليل الحالات التطبيقية، والأنشطة القصيرة المرتبطة بالمحاور اليومية. يحصل المشاركون في نهاية البرنامج على شهادة حضور/إتمام من AINFCT وفق متطلبات الحضور والمشاركة المعتمدة.

course_key_competencies

- البرمجة الآمنة.
- نمذجة التهديدات.
- مراجعة الكود الأمني.
- أمن واجهات البرمجة.
- حماية البيانات.
- أمن سلسلة التوريد البرمجية.

مكتب القاهرة الرئيسي
القاهرة، جمهورية مصر العربية
7 شارع وهران، الطيران، مدينة نصر
+201152466358
info@ainfct.com
ainfct.com
رقم التسجيل الضريبي: 472920235
مكتب مدريد الفرعي
مدريد، إسبانيا

شارع الصحة 3، وسط المدينة، 28013 مدريد
training@ainfct.com
ainfct.com