تقرير حول مشكلة تسريب المحتوى في تطبيق تعلم عن بعد والحل المقترح

مقدمة

تطبيق مخصص لتقديم الكورسات التعليمية، واجه مشكلة رئيسية تتمثل في تسريب المحتوى (الكورسات) من داخله، على الرغم من وجود نظام حماية مدمج ضد المحاكيات ووضع المطور. الهدف من هذا التقرير هو تحليل المشكلة، استعراض المحاولات التي تمت لاختراق نظام الحماية، وتقديم الحل الناجح الذي تم التوصل إليه.

وصف المشكلة

- المشكلة الأساسية: تسريب الكورسات من التطبيق على الرغم من وجود نظام حماية يمنع تشغيل التطبيق على المحاكيات أو في حالة تفعيل وضع المطور.
- التحدي: المستخدمون يجدون طرقًا لتسجيل شاشة التطبيق أو الوصول إلى المحتوى بطرق غير مصرح بها.
- الهدف: تحديد الثغرة في نظام الحماية وإيجاد حل لسدها.

المحاولات التي تم تنفيذها لاختراق الحماية

لتحديد مصدر التسريب، تم تجربة عدة طرق لتجاوز نظام الحماية المدمج في التطبيق. فيما يلي الخطوات التي تم اتخاذها:

1. استخدام المحاكيات:
   - تم تجربة محاكيات مختلفة مثل:
     - BlueStacks: محاكي شهير لتشغيل تطبيقات الأندرويد على الحاسوب.
     - Windows Subsystem for Android (WSA): نظام فرعي مدمج في ويندوز لتشغيل تطبيقات الأندرويد.
     - Android Studio: بيئة تطوير تحتوي على محاكي مدمج.
   - النتيجة: نظام الحماية في التطبيق نجح في منع التشغيل على هذه المحاكيات، مما يعني أن الحماية ضد المحاكيات فعالة.
2. تجربة فتح شاشات مشاركة متعددة (Share Screen):
   - تم افتراض أن طبقة الحماية قد تكون محددة على عرض الفيديو فقط في الشاشة الأساسية، مما قد يسمح لشاشة ثانية بعرض المحتوى بدون قيود.
   - التنفيذ: تم فتح شاشة مشاركة إضافية أثناء تشغيل التطبيق.
   - النتيجة: هذه الطريقة لم تنجح، حيث استمر نظام الحماية في منع عرض المحتوى بشكل صحيح.
3. الاستنتاج من المحاولات الأولية:
   - نظام الحماية قوي ضد المحاكيات والتلاعب المباشر بالشاشة، لكنه قد يحتوي على ثغرة في طريقة التعامل مع إعدادات الجهاز مثل وضع المطور وتسجيل الشاشة.

الحل الناجح المكتشف

بعد فشل المحاولات السابقة، تم تجربة نهج جديد يعتمد على التلاعب بإعدادات وضع المطور (Developer Mode) وتسجيل الشاشة عبر USB Debugging. الخطوات كالتالي:

1. إغلاق وضع المطور (Developer Mode):
   - تم تعطيل وضع المطور في الهاتف لضمان أن التطبيق يعمل بشكل طبيعي دون اكتشاف أي إعدادات قد تمنع تشغيله.
2. تشغيل التطبيق:
   - تم فتح تطبيق "-----" والتأكد من أنه يعمل بشكل صحيح مع عرض المحتوى المحمي.
3. إعادة تفعيل وضع المطور وUSB Debugging:
   - بعد تشغيل التطبيق، تم إعادة تفعيل وضع المطور وتوصيل الهاتف بالحاسوب عبر USB.
   - تم تفعيل خاصية USB Debugging للسماح بالتحكم في الجهاز من الحاسوب.
4. تسجيل الشاشة:
   - باستخدام أدوات مثل "scrcpy" أو أي برنامج تسجيل شاشة عبر USB، تم تسجيل شاشة الهاتف بسهولة مع عرض محتوى الكورسات.

5. النتيجة:
   - نجحت هذه الطريقة في تسجيل المحتوى بجودة عالية دون أن يكتشف التطبيق عملية التسجيل أو يوقفها.

تحليل الثغرة

- سبب الثغرة: نظام الحماية في التطبيق يعتمد على فحص حالة وضع المطور عند بدء التشغيل فقط، ولا يقوم بمراقبة التغييرات في هذه الحالة أثناء التشغيل. كما أنه لا يكتشف عمليات تسجيل الشاشة عبر USB Debugging.
- التأثير: هذا يسمح للمستخدمين بتجاوز الحماية عن طريق تشغيل التطبيق في بيئة "آمنة" ثم تغيير الإعدادات لاحقًا.

التوصيات لتحسين الحماية

لمنع تكرار هذا النوع من التسريب، يمكن تطبيق الإجراءات التالية:

1. المراقبة المستمرة لوضع المطور:
   - إضافة فحص دوري داخل التطبيق للتأكد من أن وضع المطور لم يتم تفعيله أثناء التشغيل، وإغلاق التطبيق فور اكتشاف ذلك.
2. اكتشاف USB Debugging:
   - إضافة كود للتحقق من تفعيل USB Debugging، مع إيقاف تشغيل المحتوى في حالة اكتشافه.
3. إضافة dynamic water mark
   - ظهور ال user Id فوق الفيديو مع طبقة تشفير
   - حركة مستمرة طول فترة عرض الفيديو (fade in & fade out)
   - الحركة عشوائية
4. حماية تسجيل الشاشة:
   - استخدام واجهات برمجية (APIs) مثل FLAG_SECURE في الأندرويد لمنع تسجيل الشاشة أو التقاطها.
5. تشفير المحتوى:
   - تشفير الفيديوهات داخل التطبيق بحيث لا يمكن عرضها خارج التطبيق حتى لو تم تسجيلها او تحميلها .
6. Version control
   - تقييد البرنامج بحيث يعمل على إصدارات الاندرويد 8+

الخلاصة

تم تحديد ثغرة في تطبيق "-----" تتعلق بإمكانية تسجيل الشاشة عبر تفعيل USB Debugging بعد بدء تشغيل التطبيق. الحل الناجح الذي تم اختباره يعتمد على التلاعب بوضع المطور، مما يكشف عن ضعف في تصميم نظام الحماية الحالي. التوصيات المقترحة ستساعد في تعزيز الأمان ومنع التسريب في المستقبل.

Reported by : Mostafa Nassar

Date : 12 APR 2025

Report NO.1

App Name : hidden-info