

مشروع مادة البايثون

بسم الله الرحمن الرحيم، والصلاة والسلام على أشرف الأنبياء والمرسلين سيدنا محمد وعلى آله وصحبه أجمعين.

هذا المشروع عباره عن أداة عملية بلغة **Python** تقوم بفحص الروابط (URLs) والتحقق من صحتها. تأتي أهمية المشروع من الحاجة المتزايدة في مجال الأمن السيبراني وتطوير الويب إلى أدوات تساعد في اختبار الروابط واكتشاف الروابط المعطوبة أو المشبوهة بسرعة وكفاءة.

✨ خصائص المشروع

- واجهة استخدام **CLI** ملونة واحترافية تسهل قراءة النتائج وتجعل الأداة أكثر وضوحًا.
- تقديم تقرير شامل يحتوي على معلومات أساسية عن الرابط، مستوى الأمان، والتوصية المناسبة للمستخدم.
- القدرة على كشف الروابط المختصرة التي غالبًا تُستخدم لإخفاء الوجهة الحقيقية.
- فحص وجود كلمات مشبوهة داخل الرابط قد تدل على محاولات تصيد أو نشاط غير آمن.
- نظام تقييم أمني (**Security Score**) يمنح المستخدم درجة من 100 ويصنف الرابط إلى مستويات: آمن، متوسط، خطر، أو خطير.
- إمكانية حفظ التقرير في ملف نصي لاستخدامه لاحقًا أو مشاركته ضمن التوثيق الأكاديمي.

🎯 أهداف المشروع

- مساعدة المستخدمين على التمييز بين الروابط الآمنة والمشبوهة.
- تعزيز الممارسات الأمنية في التعامل مع الروابط الإلكترونية.
- توفير أداة تعليمية وأكاديمية يمكن استخدامها في مشاريع الأمن السيبراني.

📌 القيمة الأكاديمية

هذا المشروع يجمع بين الجانب العملي والتطبيقي في الأمن السيبراني وبين الجانب الأكاديمي في تنظيم الكود وتوثيقه، مما يجعله نموذجًا متكاملًا

⚙️ آلية عمل مشروع URL Security Checker

تقوم الأداة بتنفيذ مجموعة من الخطوات المتسلسلة لفحص الرابط وإصدار تقرير أمني شامل، ويمكن تلخيص آلية العمل كما يلي:

1. إدخال الرابط

- يحصل البرنامج على الرابط من المستخدم عبر واجهة سطر الأوامر
- إذا لم يكن الرابط يحتوي على بروتوكول (http/https) ، يتم إضافة https:// تلقائيًا.

2. تحليل الرابط

- يتم استخراج معلومات أساسية مثل: اسم النطاق (Domain) والبروتوكول المستخدم (Protocol).
- التحقق من أن الرابط مكتوب بصيغة صحيحة.

3. التأكد من الوصول إلى الموقع

- إرسال طلب HTTP للتأكد من أن الموقع يعمل ويستجيب.
- معالجة الأخطاء المحتملة مثل: انتهاء المهلة، مشاكل شهادة SSL ، أو فشل الاتصال.

4. الفحص الأمني للرابط

- البحث عن كلمات مشبوهة داخل الرابط (مثل login ، bank ، password) والتي قد تدل على محاولات تصيد.

- التحقق مما إذا كان الرابط مختصرًا باستخدام خدمات مثل bit.ly أو tinyurl.

5. حساب درجة الأمان (Security Score)

- يبدأ التقييم من 100 نقطة.
- يتم خصم نقاط حسب عوامل الخطر (بروتوكول غير آمن، طول الرابط، كلمات مشبوهة، روابط مختصرة).
- تصنيف الرابط إلى مستويات: آمن، متوسط، خطر، أو خطير.

6. إصدار التقرير النهائي

- عرض النتائج بشكل ملون ومنظم في واجهة CLI.
- تضمين معلومات أساسية، درجة الأمان، مستوى الحماية، والتوصية للمستخدم.
- إمكانية حفظ التقرير في ملف نصي عند الحاجة.

🌟 الخلاصة

آلية عمل المشروع تعتمد على تحليل الرابط، التحقق من صلاحيته، فحص عوامل الخطر، ثم إصدار تقرير أمني شامل يساعد المستخدم على اتخاذ قرار آمن عند التعامل مع الروابط الإلكترونية.

🌟 توزيع المهام في مشروع URL Security Checker

لقد تم تنفيذ المشروع بروح الفريق الواحد، حيث تكاملت جهودنا لتحقيق النتيجة النهائية بشكل احترافي ومنظم:

- زياد سالم: تولى مسؤولية كتابة الأكواد البرمجية بلغة Python ، حيث قام بتصميم وتنفيذ السكربت الأساسي للأداة، مع التركيز على الجوانب الأمنية

والوظائف العملية، إضافةً إلى الاهتمام بجمالية واجهة CLI الملونة وتوثيق خطوات العمل بشكل منهجي.

- **زياد الحبابي:** قام بمهمة رفع المشروع على منصة **GitHub** وتجهيزه للنشر الأكاديمي، كما تولى إعداد ملف **PDF** الرسمي للتوثيق الذي يشرح تفاصيل المشروع، أهدافه، وخصائصه، مما ساهم في إظهار العمل بصورة احترافية أمام الدكتور