SkyGuard

دليل أوامر التشغيل والفحص: ماذا استخدمنا، ولماذا استخدمناه

هذا الملف يشرح الأوامر التي استُخدمت أثناء تجهيز تطبيق SkyGuard وتشغيله وتطوير فحص Nmap. الهدف أن تكون قادرًا على فهم كل أمر بدل تشغيله كأمر غامض، وأن تعرف الفرق بين الفحص السريع والفحص الأدق.

المشروع: SkyGuard - Security Scanner

المنصة: Windows + Python + PyQt6 + Nmap

نوع الفحص: فحص حقيقي باستخدام برنامج Nmap المثبت على النظام

المخرجات: جداول نتائج + تقرير أمني قابل للتصدير

1. أوامر تجهيز وتشغيل التطبيق

الأمر	لماذا استخدمناه؟
`pip install -r requirements.txt`	يثبت مكتبات Python المطلوبة لتشغيل SkyGuard مثل PyQt6 للواجهة، وpython-nmap للتعامل مع Nmap، وrequests لفحص الدومينات، وgoogle-generativeai للتقرير الذكي.
`python main.py`	يشغل تطبيق SkyGuard. ملف `main.py` هو نقطة الدخول التي تفتح واجهة PyQt6.
`.\.venv\Scripts\python.exe main.py`	يشغل التطبيق باستخدام البيئة الافتراضية الخاصة بالمشروع. هذا أفضل من الاعتماد على Python عام في النظام، لأن كل المكتبات تكون مثبتة داخل المشروع نفسه.
`python -m compileall .`	استخدمناه كاختبار سريع للتأكد أن ملفات Python لا تحتوي أخطاء تركيبية تمنع التشغيل.

مهم: مكتبة `python-nmap` لا تكفي وحدها. يجب أن يكون برنامج Nmap نفسه مثبتًا على Windows، لأن المكتبة فقط ترسل الأوامر إلى `nmap.exe` وتقرأ النتائج.

2. أوامر تثبيت واكتشاف Nmap

الأمر	الشرح
`winget install -e --id Insecure.Nmap`	يثبت Nmap من مدير الحزم winget. استخدمنا -e للمطابقة الدقيقة لاسم الحزمة.
`winget install -e --id Insecure.Nmap --accept-source-agreements --accept-package-agreements`	نفس أمر التثبيت، لكن مع قبول الاتفاقيات بشكل غير تفاعلي حتى لا يتوقف الأمر عند سؤال داخل الطرفية.
`Get-Command nmap`	يتحقق هل أمر nmap متاح مباشرة من PowerShell عبر متغير PATH. عندك لم يكن متاحًا، لذلك التطبيق لم يجده أولًا.
`C:\Program Files (x86)\Nmap\nmap.exe --version`	يتحقق من أن Nmap مثبت فعلًا ويعرض رقم الإصدار. النتيجة عندك أظهرت Nmap 7.99.

3. أمر الفحص الحقيقي الذي اختبرناه

```
C:\Program Files (x86)\Nmap\nmap.exe -sV --version-light -T4 -Pn --top-ports 100 192.168.1.1
```

هذا هو فحص Nmap الحقيقي الذي أثبت أن النتائج ليست وهمية. الفحص رجّع أن الجهاز 192.168.1.1 يعمل ووجد منافذ مفتوحة مثل 25 و110 و143 و465 و587 و993 و995.

شرح خيارات الأمر

الخيار	معناه	سبب استخدامه
`-sV`	تفعيل كشف الخدمة والإصدار.	حتى لا نعرف فقط أن المنفذ مفتوح، بل نحاول معرفة نوع الخدمة مثل SMTP أو HTTP.
`--version-light`	كشف إصدارات أخف وأسرع.	يسرع الفحص مقارنة بـ `--version-all` مع المحافظة على معلومات مفيدة.
`-T4`	توقيت أسرع للفحص.	مناسب للشبكات المحلية غالبًا، ويقلل وقت الانتظار.
`-Pn`	اعتبار الهدف موجودًا بدون الاعتماد على ping.	بعض الراوترات والجدران النارية تمنع ping، فيظهر الهدف كأنه غير موجود رغم أنه يعمل.
`--top-ports 100`	فحص أكثر 100 منفذ شيوعًا.	يعطي نتيجة أسرع من فحص كل المنافذ، ومفيد كبداية عملية.

تحذير Npcap: ظهر عندك تحذير أن Nmap لم يستطع تحميل كل وظائف Npcap. هذا لا يعني أن الفحص وهمي؛ Nmap استخدم وضع `connect()`. لكن تحديث Npcap يحسن الدقة والقدرات على Windows.

4. أوضاع الفحص الجديدة داخل SkyGuard

الوضع	الأمر الداخلي	متى تستخدمه؟
Fast	`-sT -sV --version-light -T4 -Pn --top-ports 100 --max-retries 2`	للحصول على نتيجة سريعة وفحص أولي للشبكة أو الراوتر.
Balanced	`-sT -sV --version-light -T3 -Pn --top-ports 1000 --max-retries 3`	للتوازن بين السرعة والدقة، ويفحص منافذ أكثر.
Accurate	`-sT -sV --version-all -T3 -Pn --top-ports 1000 --max-retries 3`	عندما تريد معلومات أدق عن الخدمات والإصدارات، لكنه أبطأ.
Full TCP	`-sT -sV --version-light -T3 -Pn -p 1-65535 --max-retries 2`	لفحص كل منافذ TCP. استخدمه فقط عند الحاجة لأنه قد يستغرق وقتًا طويلًا.

خيارات إضافية مهمة

الخيار	المعنى	السبب
`-sT`	TCP Connect scan.	أكثر توافقًا على Windows، خصوصًا عند وجود مشكلة أو نقص في Npcap.
`--max-retries 2`	تقليل عدد إعادة المحاولة.	يزيد السرعة، خصوصًا عندما تكون المنافذ filtered ولا ترد.
`--max-retries 3`	إعادة محاولات أكثر قليلًا.	يزيد موثوقية الفحص في الوضع المتوازن أو الدقيق.
`-p 25,80,443`	فحص منافذ محددة فقط.	أسرع طريقة عندما تعرف المنافذ التي تريد اختبارها.
`-p 1-1000`	فحص نطاق منافذ.	مفيد عندما تريد فحص نطاق معين بدل كل منافذ TCP.

5. أوامر إصلاح البيئة الافتراضية

الأمر	لماذا استخدمناه؟
`.\.venv\Scripts\python.exe -m ensurepip --upgrade`	أصلحنا به pip داخل البيئة الافتراضية لأن جزءًا منه كان مفقودًا، وظهر خطأ `ModuleNotFoundError: No module named ''pip._internal.utils`.
`.\.venv\Scripts\python.exe -m pip install -r requirements.txt`	ثبتنا المتطلبات داخل `.venv` حتى يعمل التطبيق بنفس البيئة التي يستخدمها VS Code.
`.\.venv\Scripts\python.exe -c "from core.scanner import NetworkScanner; ..."`	اختبار مباشر لمحرك الفحص بدون فتح الواجهة. استخدمناه للتأكد أن الكود يستدعي Nmap الحقيقي ويقرأ النتائج.

6. كيف تختار الفحص المناسب؟

- **تريد نتيجة بسرعة؟** استخدم Fast.
- **تريد فحصًا يوميًا جيدًا؟** استخدم Balanced.
- **تريد دقة أعلى في الخدمات والإصدارات؟** استخدم Accurate.
- **تبحث عن منفذ غير شائع؟** استخدم Full TCP أو اكتب نطاقًا مخصصًا في خانة Ports.
- **تريد اختبار منافذ محددة؟** اكتبها مثل `25,80,443` وسيستخدم SkyGuard هذه المنافذ بدل قائمة top ports.

7. ملاحظات أمنية مهمة

SkyGuard لا يفترض وجود ثغرة بدون دليل. وجود منفذ مفتوح يعني أن هناك خدمة يمكن مراجعتها، لكنه لا يعني تلقائيًا أن الجهاز مخترق أو أن الخدمة ضعيفة.

التحليل الصحيح يعتمد على: المنفذ، الخدمة، الإصدار، هل الخدمة مكشوفة للإنترنت أم داخل الشبكة، وهل توجد سياسات وصول أو جدار ناري.

تم إنشاء هذا الدليل لمشروع SkyGuard المحلي. استخدم الفحص فقط على أجهزتك أو الشبكات التي تملك إذنًا قانونيًا لفحصها.