Legion auf Kali Linux – Installations- & Betriebsdokumentation

Root + venv · rockyou-Einbindung · Nmap/Parser-Fixes · Troubleshooting

---

Stand: 12.08.2025

Zielgruppe: Pentesting/Blue Team · Schulung & Betrieb

Hinweis: Nur in autorisierten Netzwerken einsetzen.

Inhaltsverzeichnis

1) Zweck & Überblick

Legion ist ein GUI-Tool fuer Netzwerk-Recon und Schwachstellenscans (Nmap, Hydra u. a.). Diese Anleitung zeigt Installation und Betrieb als Root in einer Python-venv, bindet die Passwortliste rockyou ein und enthaelt Workarounds/Fixes fuer bekannte Parser-Crashes.

2) Voraussetzungen

Kali Linux (rolling) · Root-Shell (sudo su -) · Internet · ~300 MB frei

3) Installation (Root + venv)

```
sudo su -
apt update && apt full-upgrade -y
apt install -y git python3 python3-pip python3-venv python3-pyqt5 \
               python3-sqlalchemy python3-lxml nmap hydra nikto whatweb \
               seclists dirb
apt install -y python3.11 python3.11-venv || true
cd /root
git clone https://github.com/GoVanguard/legion.git || true
cd /root/legion
python3.11 -m venv venv 2>/dev/null || python3 -m venv venv
source venv/bin/activate
pip install --upgrade pip
pip install -r requirements.txt
chmod +x legion.py
```

4) Start & Komfortstarter

```
export LANG=C
cd /root/legion && source venv/bin/activate
python3 legion.py

cat >/usr/local/bin/legion-venv <<'SH'
#!/usr/bin/env bash
export LANG=C
cd /root/legion || exit 1
source venv/bin/activate
exec python3 legion.py
SH
chmod +x /usr/local/bin/legion-venv
```

5) rockyou-Passwortliste einbinden

Die Datei /usr/share/wordlists/rockyou.txt.gz entpacken:

```
gzip -dk /usr/share/wordlists/rockyou.txt.gz
```

5.1) Nutzung in Hydra (Legion-GUI)

Im Hydra-Dialog Password list (-P) auf /usr/share/wordlists/rockyou.txt setzen; Threads moderat waehlen.

```
hydra -l root -P /usr/share/wordlists/rockyou.txt -t 4 ssh://192.168.1.10
```

5.2) Nutzung in Nmap-NSE-Brute-Skripten

Script-Args userdb/ passdb setzen und XML-Ausgabe verwenden:

```
nmap -sV -p22 --script ssh-brute \
  --script-args userdb=/root/users.txt,passdb=/usr/share/wordlists/rockyou.txt \
  -oX - 192.168.1.10
```

6) Empfohlene Nmap-Vorlage (stabil)

```
nmap -sS -sV -O -T4 --script "default,safe" -oX -
nmap --script-updatedb
chmod u+s $(which nmap)
```

7) Troubleshooting

7.1) IndexError in processVulnersScriptOutput (CPE-Version)

Sofort ohne vulners laufen lassen; empfohlenen Patch anwenden:

```
cat >/root/patch_legion_vulners_v2.sh <<'SH'
#!/usr/bin/env bash
set -euo pipefail
FILE="/root/legion/legion/parsers/Script.py"
[[ -f "$FILE" ]] || { echo "[!] $FILE fehlt"; exit 1; }
cp -a "$FILE" "${FILE}.bak.$(date +%F_%H%M%S)"
python3 - <<'PY'
import re, pathlib
p = pathlib.Path("/root/legion/legion/parsers/Script.py")
s = p.read_text(encoding="utf-8")
s_new = re.sub(
    r"resultCpeDetails\['version'\]\s*=\s*resultCpeData\[4\]",
    "resultCpeDetails['version'] = (resultCpeData[4] if len(resultCpeData) > 4 and resultCpeD
    s
)
if s_new == s:
    print('Pattern nicht gefunden – bitte Stelle pruefen.')
else:
    p.write_text(s_new, encoding="utf-8")
    print('OK: Patch angewendet.')
PY
SH
chmod +x /root/patch_legion_vulners_v2.sh
bash /root/patch_legion_vulners_v2.sh
```

7.2) AttributeError in NmapImporter (db_script/scr None)

Defensiver try/except-Ersatz fuer db_script.output-Zuweisung:

```
cat >/root/patch_legion_nmapimporter_try_except_v2.sh <<'SH'
#!/usr/bin/env bash
set -euo pipefail
FILE="/root/legion/legion/app/importers/NmapImporter.py"
[[ -f "$FILE" ]] || { echo "[!] $FILE fehlt"; exit 1; }
cp -a "$FILE" "${FILE}.bak.$(date +%F_%H%M%S)"
python3 - <<'PY'
import re, pathlib, sys
p = pathlib.Path("/root/legion/legion/app/importers/NmapImporter.py")
s = p.read_text(encoding="utf-8")
```

```
pat = re.compile(r'^([ \t]*)db_script\s*\.\s*output\s*=\s*(?:getattr\(\s*scr\s*,\s*["\']outpu
def repl(m):
    ind = m.group(1)
    return f'{ind}try:\n{ind}    db_script.output = getattr(scr, \"output\", \"\")\n{ind}exce
s2, n = pat.subn(repl, s)
if n == 0:
    print('Pattern nicht gefunden – bitte manuell pruefen.')
    sys.exit(2)
p.write_text(s2, encoding='utf-8')
print(f'OK: Patch an {n} Stelle(n).')
PY
SH
chmod +x /root/patch_legion_nmapimporter_try_except_v2.sh
bash /root/patch_legion_nmapimporter_try_except_v2.sh
```

8) Pflege & Updates

```
cd /root/legion
git pull
source venv/bin/activate
pip install -r requirements.txt
nmap --script-updatedb
```

9) Sicherheit & Rechtliches

Scans/Bruteforce nur mit Genehmigung. Sorgfalt bei Root-Tools. Ergebnisse vertraulich behandeln.