👻 Scarlet Ghost: Documentação Oficial

Versão: 5.0 (Ultimate)

Autor: CyberGhost

1. Visão Geral e Propósito

O **Scarlet Ghost** é um framework de automação em Bash projetado para **Pentesting** (Testes de Intrusão) e **Bug Bounty**.

Para que serve no dia a dia?

No cotidiano de um profissional de segurança ofensiva, configurar um ambiente de trabalho leva horas. É necessário instalar linguagens (Go, Python, Ruby), configurar variáveis de ambiente e baixar dezenas de ferramentas de repositórios diferentes.

O Scarlet Ghost resolve isso:

1. **Padronização:** Garante que todas as ferramentas (Nuclei, Amass, Nmap, etc.) estejam instaladas nas versões corretas.
2. **Manutenção:** Atualiza o sistema e as ferramentas automaticamente.
3. **Dependências:** Resolve problemas chatos de bibliotecas (como libpcap para o Naabu ou ferramentas Python).
4. **Análise Rápida:** Possui um módulo embutido para processar logs de DNS e gerar relatórios HTML.

2. Estrutura de Diretórios

Ao ser executada, a ferramenta cria uma estrutura oculta na home do usuário para manter a organização:

- ~/.scarlet-ghost/ → Diretório raiz.
- ├── logs/ → Registros de tudo que foi instalado ou erros ocorridos.
- ├── templates/ → Modelos para o Nuclei (padrão e customizados).
- ├── wordlists/ → Listas de senhas e diretórios (SecLists, FuzzDB).
- ├── output/ → Onde os relatórios gerados são salvos.
- ├── backups/ → Backups das configurações da ferramenta.
- └── tools-installed.json → Banco de dados local do status das ferramentas.

3. Detalhamento do Menu de Opções

Abaixo, a explicação técnica do que cada opção do menu faz no sistema operacional:

📦 Instalação e Core

- **1) Instalação Completa:**
  - Executa todas as funções de instalação em sequência. Ideal para VPS novas ou máquinas recém-formatadas. Verifica internet, disco, memória e instala Go, Python, e todas as categorias de ferramentas abaixo.
- **2) Nuclei Suite:**
  - Foca no ecossistema da ProjectDiscovery. Instala nuclei, subfinder, httpx, dnsx.
  - **Diferencial:** Instala automaticamente a biblioteca libpcap-dev, necessária para o naabu (scanner de portas) funcionar sem erros, algo que falha em instalações manuais.
- **3) Tomnomnom Tools:**
  - Instala as ferramentas lendárias do pesquisador Tomnomnom (waybackurls, assetfinder, gf).
  - **Configuração Extra:** Baixa e configura automaticamente os "GF Patterns" (padrões de regex) em ~/.gf para filtrar vulnerabilidades como XSS, SQLi e SSRF.

🛠️ Categorias de Ferramentas

- **4) Python Tools:**
  - Configura o pip3 e instala libs essenciais (scapy, impacket, frida).
  - Clona e instala ferramentas que não estão no pip, como SQLMap, XSStrike e ParamSpider direto do GitHub.

- **5) Recon Tools:**
  - Ferramentas para mapeamento de superfície de ataque. Inclui Amass, MassDNS e

Gau.
- **6) Web Tools:**
  - Focado em ataque web direto. Instala FFUF (fuzzing), WPScan (WordPress) e Nikto.

📊 Utilidades e Gestão

- **7) Wordlists:**
  - Baixa gigabytes de listas essenciais: SecLists, FuzzDB e PayloadsAllTheThings.
- **8) Verificar Tools:**
  - Roda um script de auditoria que checa se os binários estão acessíveis no $PATH do sistema. Mostra uma barra de progresso visual.
- **9) Exportar Lista:**
  - Gera um arquivo .txt listando todas as ferramentas detectadas e seus caminhos de instalação. Útil para relatórios de auditoria do ambiente.

🔄 Sistema e Processamento

- **10) Update Sistema:**
  - Detecta automaticamente o gerenciador de pacotes (apt, pacman, dnf, brew) e roda os comandos de atualização e limpeza de cache.
- **11) Processar DNS (Módulo Avançado):**
  - Lê um arquivo JSON (gerado por ferramentas como dnsx -json).
  - Separa os registros por tipo (A, AAAA, CNAME, MX).
  - Gera estatísticas e um relatório HTML se o pandoc estiver instalado.
- **12) Backup Config & 13) Restore Backup:**
  - Cria/Restaura um arquivo .tar.gz contendo toda a pasta .scarlet-ghost. Útil para migrar configurações entre máquinas.
- **14) WSL Check:**
  - Verifica se você está rodando no "Windows Subsystem for Linux", o que é útil pois algumas ferramentas de rede de baixo nível podem precisar de ajustes no WSL.

4. Passo a Passo: Guia de Uso (Comandos de Teste)

Como o Scarlet Ghost é um **instalador e gerenciador**, o "teste" consiste em usar as ferramentas que ele instalou.

Cenário: Reconhecimento em um alvo fictício empresa-teste.com

Passo 1: Preparação

Primeiro, execute o script para garantir que tudo está instalado.

Bash

```
chmod +x scarlet-ghost.sh
sudo ./scarlet-ghost.sh
# Selecione a opção 1 (Instalação Completa) e aguarde.
```

Passo 2: Reconhecimento de Subdomínios (Usando ferramentas instaladas)

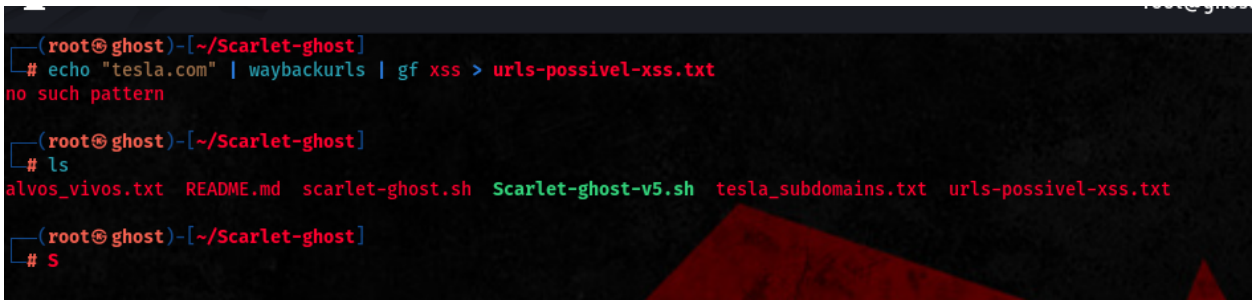
Abra um novo terminal (para recarregar o PATH) e use o subfinder instalado pela opção 2.

Bash

```
subfinder -d empresa-teste.com -o subdominios.txt
```

Passo 3: Filtragem de URLs (Usando ferramentas da opção 3)

Vamos pegar URLs antigas arquivadas e filtrar as que parecem interessantes.



Bash

```
echo "empresa-teste.com" | waybackurls | gf xss > urls-possivel-xss.txt
```

Passo 4: Escaneamento de Portas e Serviços (Usando ferramentas da opção 2)

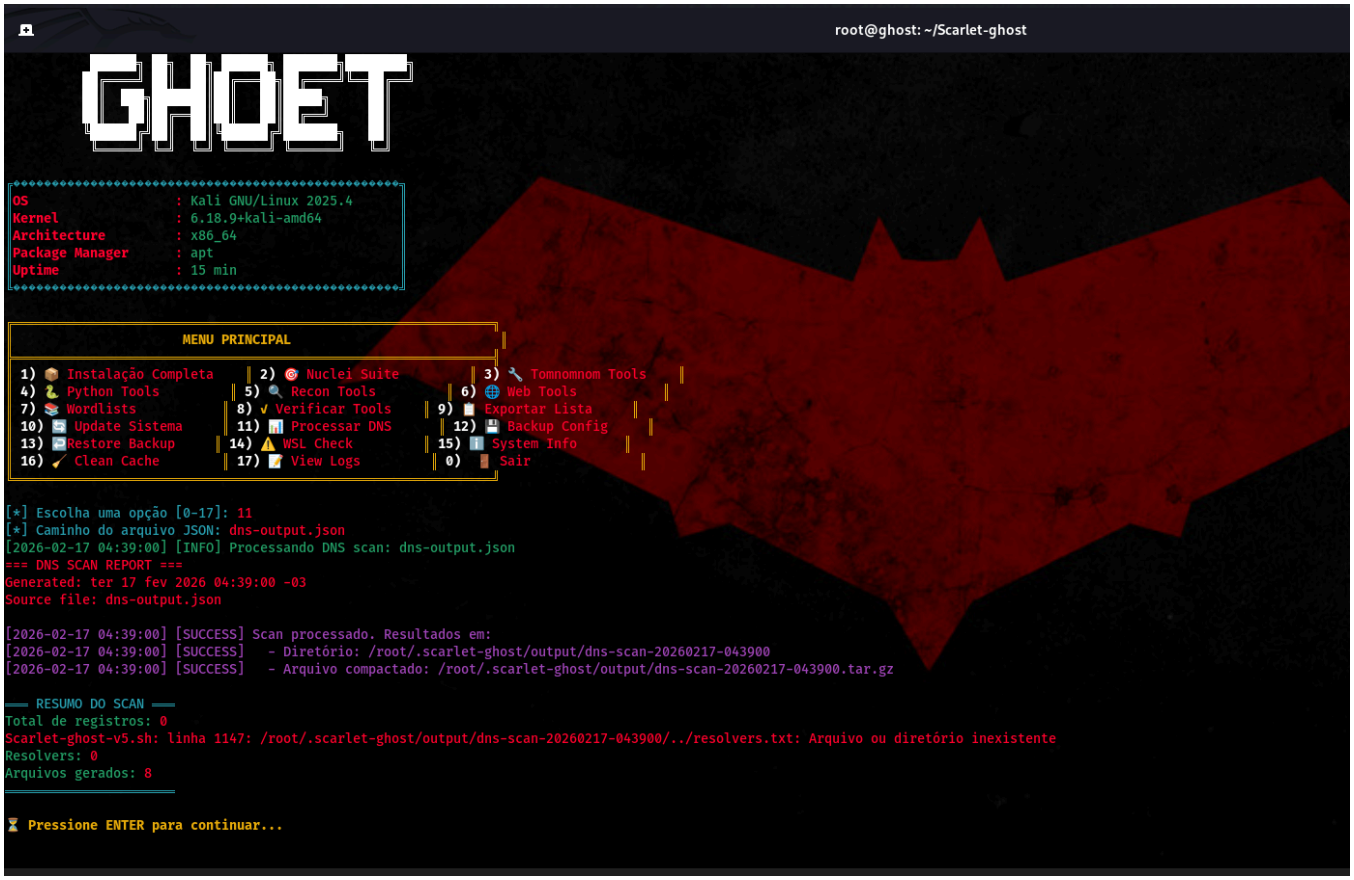
Vamos gerar um JSON de DNS para testar a **Opção 11** do menu.

Bash

```
cat subdominios.txt | dnsx -resp -json -o dns-output.json
```

Passo 5: Utilizando o Processador Interno (Opção 11)

1. Execute ./scarlet-ghost.sh novamente.
2. Escolha a opção **11) Processar DNS**.
3. Quando pedir o caminho, digite: dns-output.json.
4. O script vai gerar uma pasta em ~/.scarlet-ghost/output/ com arquivos separados (ex: a_records.txt, cname_records.txt) e um relatório.



5. Comandos e Configurações Avançadas

A. Templates Customizados do Nuclei

O script cria automaticamente um modelo de exemplo em ~/.scarlet-ghost/templates/custom-templates/default-login.yaml.

Você pode criar seus próprios templates de detecção de vulnerabilidade e salvá-los nesta pasta. O Scarlet Ghost organiza isso para que você não misture com os templates oficiais que são atualizados frequentemente.

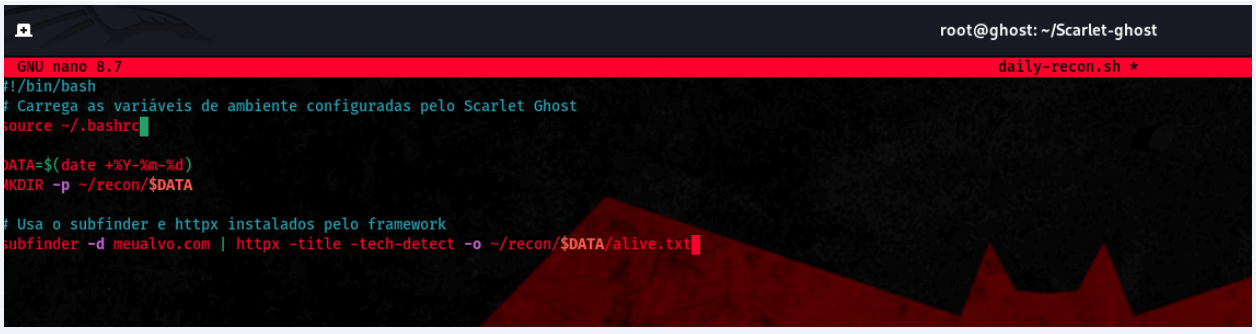
B. Automação via Cron (Agendamento)

Você pode criar um script que usa as ferramentas instaladas pelo Scarlet Ghost para rodar todo dia às 3 da manhã.

Crie um arquivo daily-recon.sh:

Bash

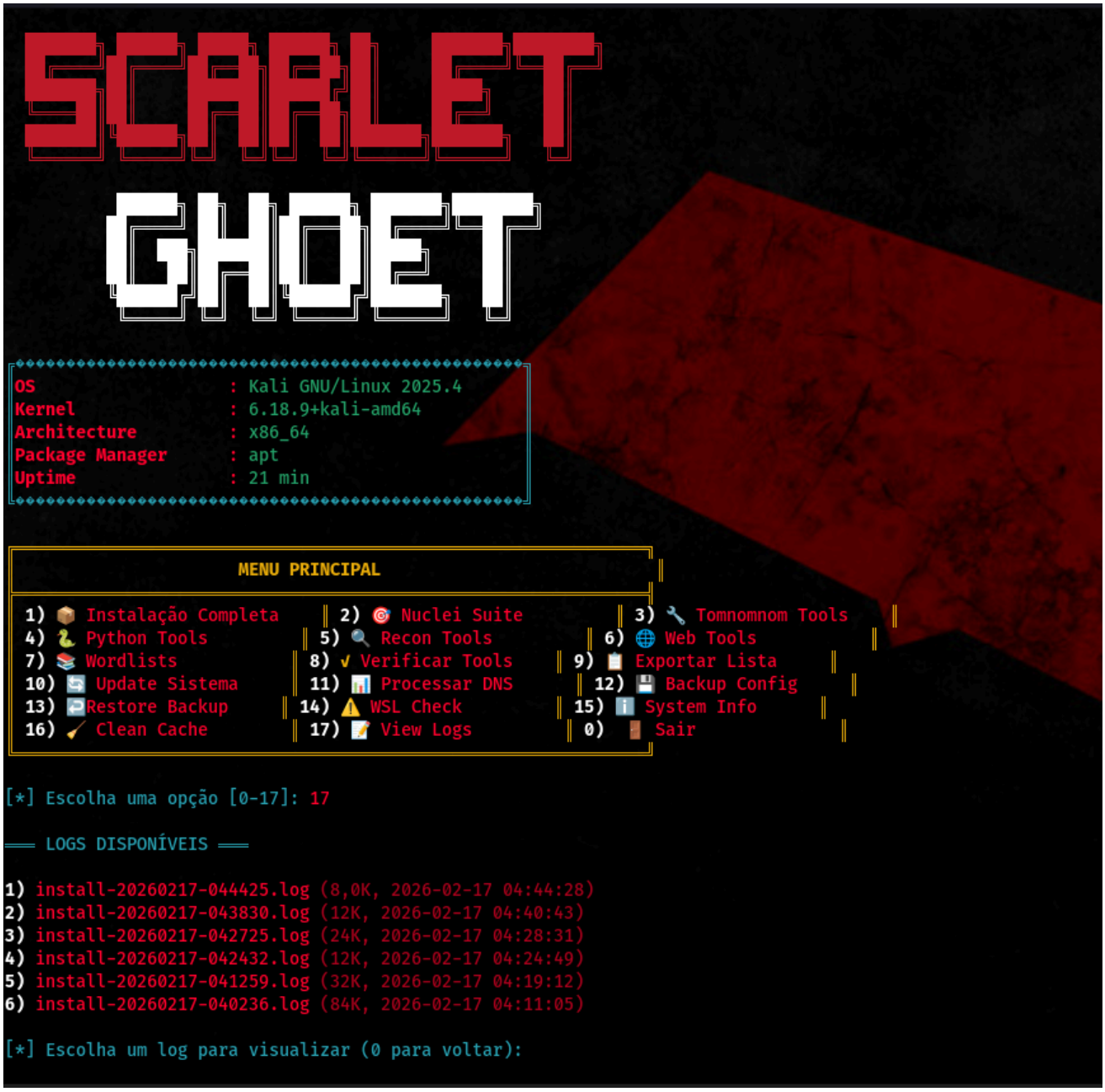
```
#!/bin/bash
# Carrega as variáveis de ambiente configuradas pelo Scarlet Ghost
source ~/.bashrc

DATA=$(date +%Y-%m-%d)
MKDIR -p ~/recon/$DATA

# Usa o subfinder e httpx instalados pelo framework
subfinder -d meualvo.com | httpx -title -tech-detect -o ~/recon/$DATA/alive.txt
```



C. Logs e Debugging

Se uma ferramenta falhar na instalação, não precisa adivinhar o erro.

1. Vá ao menu e escolha **17) View Logs**.
2. Selecione o log mais recente.
3. Procure por linhas marcadas com [ERROR]. O script mantém um histórico rotativo dos últimos 5 logs para economizar espaço.



D. Modo "Sem Root"

Embora o script peça root (sudo) para instalar pacotes do sistema (apt, pacman), a maioria das ferramentas Go e Python são instaladas na pasta do usuário (/home/user/go/bin ou ~/.local/bin).

Se você não tiver root, o script avisa, mas tenta continuar instalando o que for possível no modo usuário.

Resumo Técnico para SysAdmins

Componente	Detalhe
Linguagem	Bash (Shell Script)
Compatibilidade	Linux (Debian, Arch, Fedora, Alpine), macOS, WSL
Gerenciamento de Logs	Rotação automática (Max 10MB), Níveis de severidade (INFO, WARN, ERROR)
Resiliência	Funções com retries (tentativas) automáticas para downloads falhos
Clean Up	Função 16 limpa cache do Go, Pip, Npm e Apt para liberar espaço em disco