Hackers do Bem – Fundamental
Prof. Fábio Carneiro de Castro
12/02/2026

Atividade Prática – Módulo 2

Aulas 1 e 2

Gabriel dos Santos Schmitz

Este documento foi criado usando LaTeX
https://github.com/gabrielzschmitz/uni/tree/main/hackers-do-bem/fundamental/modulo2/aulas1-2.

1 Introdução

Este documento apresenta as evidências práticas das atividades do Módulo 2 (Aulas 1 e 2) do Programa Hackers do Bem – Nível Fundamental, por meio dos prints solicitados, demonstrando a correta execução das tarefas propostas.

Conforme orientações do curso, este documento reúne, em um único arquivo PDF, os seguintes prints obrigatórios: Atividade 2.1 (passo 15), Atividade 2.2 (passo 6), Atividade 2.3 (passo 10), Atividade 2.4 (passo 6) e Atividade 2.5 (passo 7), todos acompanhados de breve descrição explicativa para facilitar a avaliação pelo instrutor.

2 Atividades

Atividade 2.1. Criando um Trojan de Acesso Remoto com o Social-Engineer Toolkit no Kali Linux

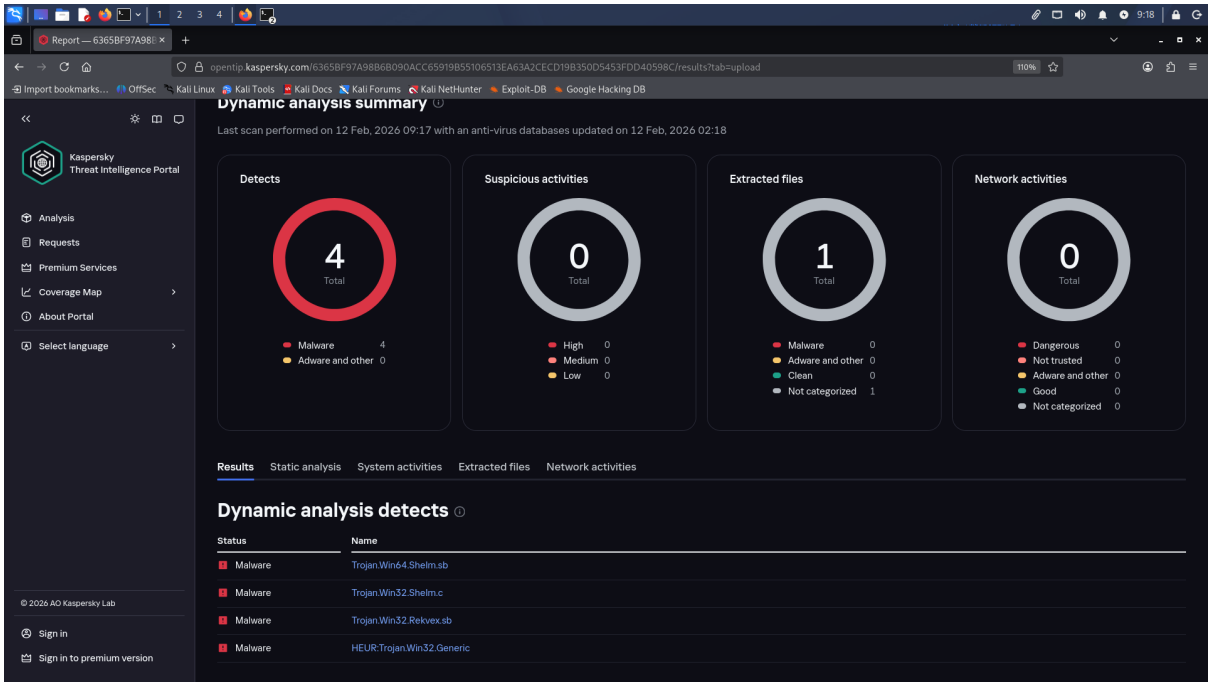

Fig. 1: Geração do payload com SEToolkit e detecção de malware no Kaspersky Threat Intelligence Portal

Sobre:

Nesta atividade, foi utilizada a ferramenta *Social-Engineer Toolkit (SET)*, nativa do Kali Linux, para gerar um *payload* do tipo **Windows Meterpreter Reverse_TCP X64**, simulando a criação de um Trojan de Acesso Remoto (RAT) para fins exclusivamente acadêmicos.

Após a execução do comando `setoolkit` com privilégios de superusuário, foram selecionadas as opções:

- **Social-Engineering Attacks**;
- **Create a Payload and Listener**;
- **Windows Meterpreter Reverse_TCP X64**.

Foi configurado o endereço IP local (`LHOST 192.168.98.40`) e a porta de escuta reversa (`LPORT 7777`). O SET gerou automaticamente o arquivo executável `payload.exe`, armazenado no diretório `/root/.set/`.

Em seguida, o *listener* foi iniciado via integração com o *Metasploit Framework*, utilizando o módulo `multi/handler`, responsável por aguardar conexões reversas provenientes da máquina alvo.

Para fins de análise de segurança, o arquivo `payload.exe` foi submetido ao *Kaspersky Threat Intelligence Portal*, onde foi detectado como malware (nível de detecção 6), confirmando seu comportamento potencialmente malicioso.

Atividade 2.2. Explorando o Keylogger XSPY no Kali Linux

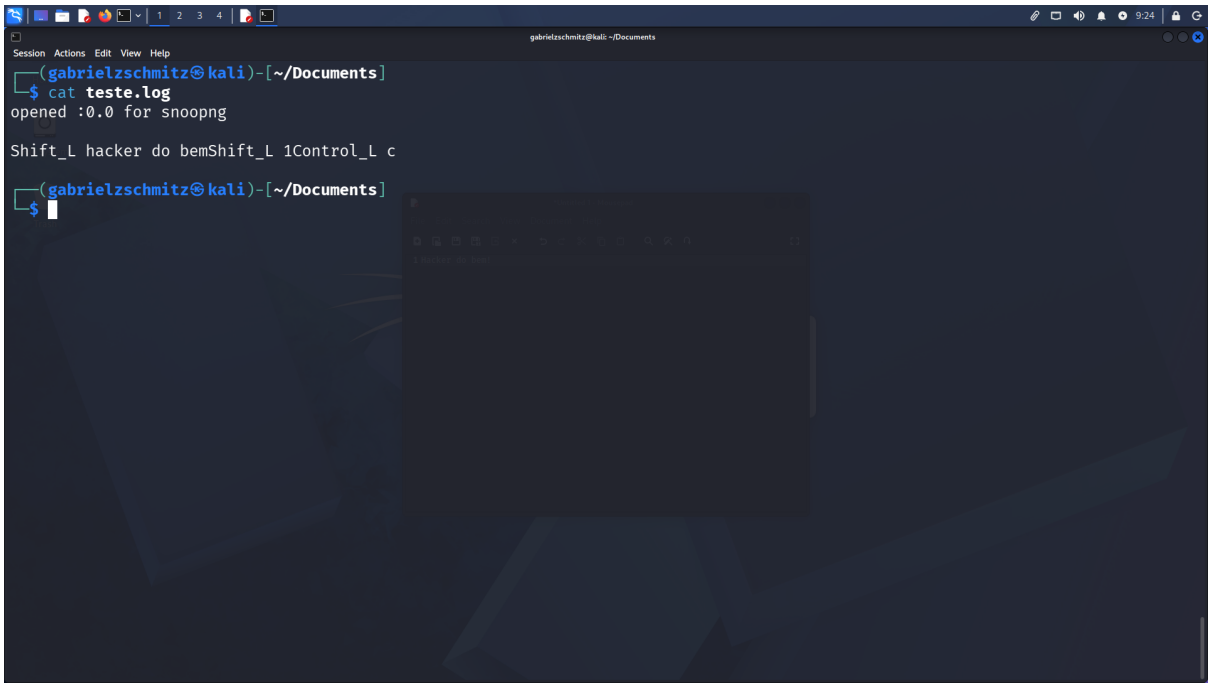

Fig. 2: Captura de teclas utilizando o XSPY e visualização do arquivo teste.log

Sobre:

Nesta atividade, foi explorada a ferramenta `xspy`, disponível no Kali Linux, para demonstrar o funcionamento de um *keylogger*, programa capaz de registrar teclas digitadas no sistema.

Inicialmente, foi acessada a pasta `/home/aluno/Documentos`, confirmando que estava vazia. Em seguida, o keylogger foi executado com o redirecionamento de saída para o arquivo `teste.log` por meio do comando:

```
xspy >> teste.log
```

Após a inicialização da ferramenta, foi aberto o Editor de Texto e digitada manualmente a frase *"Hacker do bem!"*. Posteriormente, a execução do `xspy` foi encerrada com `Ctrl + C`.

Ao visualizar o conteúdo do arquivo `teste.log` utilizando o comando `cat teste.log`, foi possível observar o registro das teclas capturadas durante a execução do keylogger, comprovando o funcionamento da ferramenta.

Por fim, o arquivo foi removido com o comando `rm teste.log`, restabelecendo o ambiente inicial.

Atividade 2.3. Explorando Ransomwares no Kali Linux

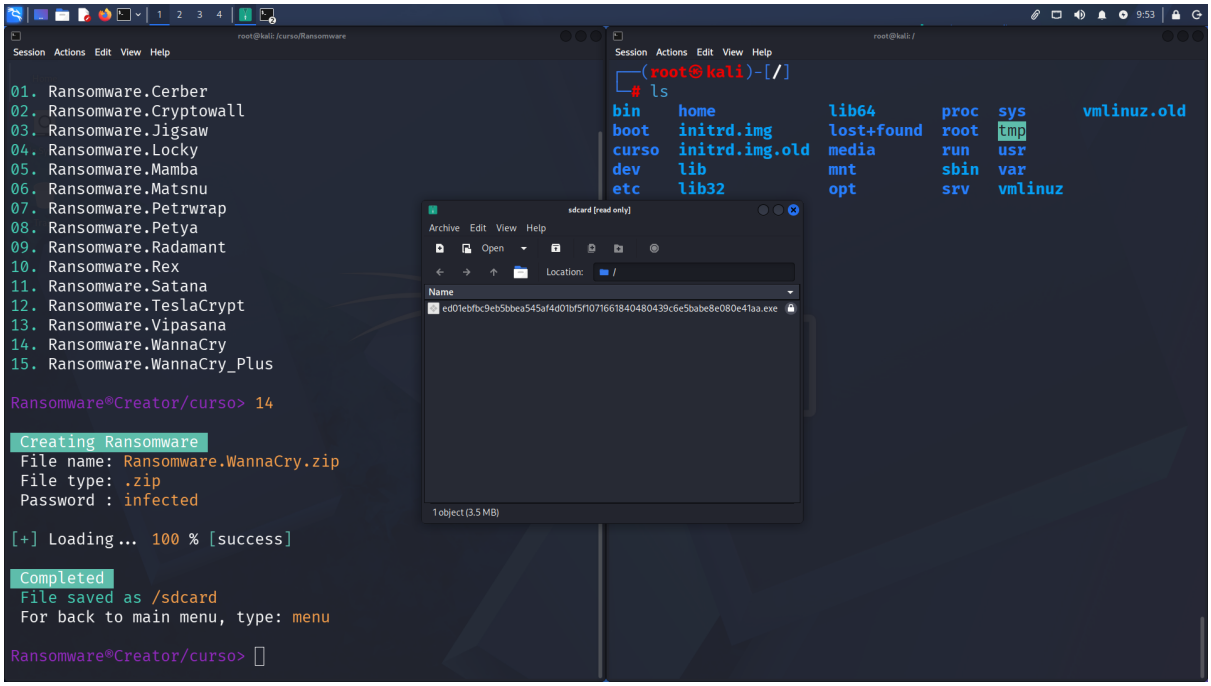

Fig. 3: Criação do Ransomware WannaCry e visualização do arquivo executável gerado

Sobre:

Nesta atividade, foi explorado um repositório educacional de criação de Ransomwares no Kali Linux, com foco na compreensão teórica do funcionamento dessas ameaças. Ressalta-se que todos os procedimentos foram realizados exclusivamente para fins acadêmicos, sem execução do código malicioso.

Após acessar o diretório `/curso/Ransomware` e executar o programa `python3 Ransomware`, foi possível visualizar o menu interativo com diferentes famílias de ransomware disponíveis para geração.

Por meio do comando `show`, foram listadas diversas variantes, incluindo Cerber, Locky, Petya e WannaCry. Em seguida, foi selecionada a opção **14 – Ransomware.WannaCry**, resultando na criação de um arquivo compactado protegido por senha (`infected`), salvo como `sdcard` no diretório raiz.

Ao acessar o arquivo pelo gerenciador de arquivos, verificou-se que seu conteúdo incluía um executável (`.exe`) com aproximadamente 3.5MB, correspondente ao ransomware gerado. O arquivo não foi executado, mantendo-se a atividade apenas no âmbito de análise estrutural.

Por fim, o arquivo `sdcard` foi removido do sistema, restaurando o ambiente inicial.

Atividade 2.4. Explorando múltiplos Payloads de Malware no Kali Linux

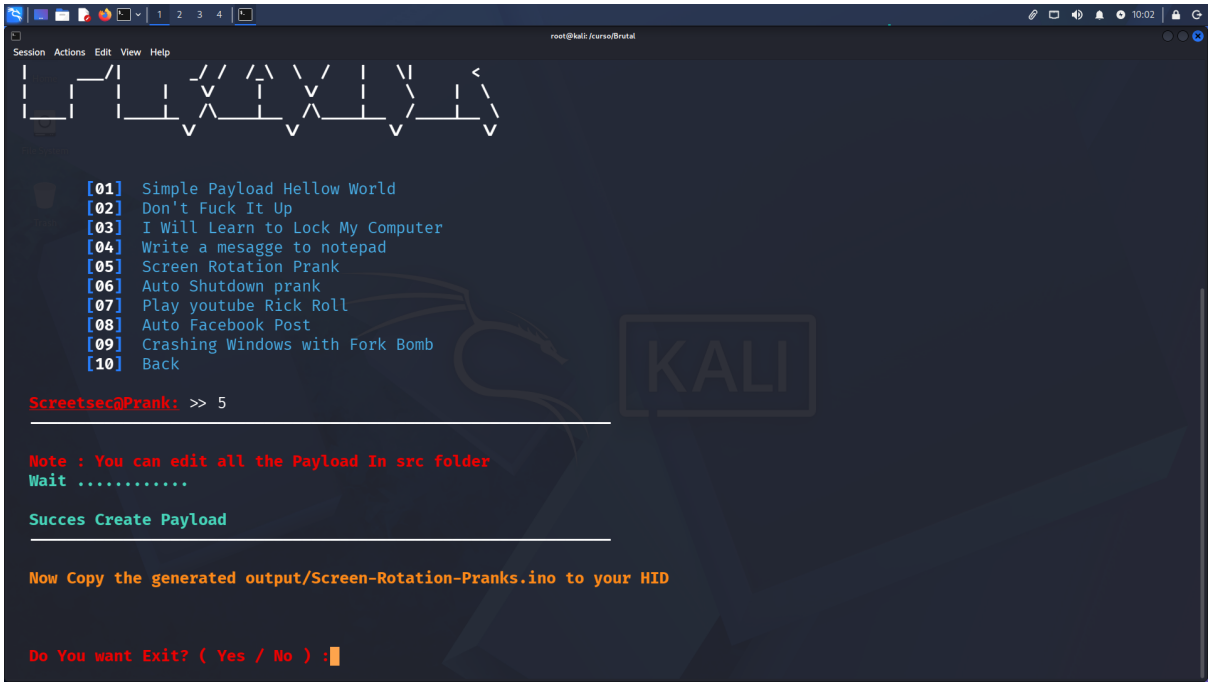

Fig. 4: Execução do Brutal.sh e geração do payload Screen-Rotation-Pranks.ino

Sobre:

Nesta atividade, foi explorado o repositório educacional `Brutal`, disponível no diretório `/curso/Brutal`, com o objetivo de compreender a estrutura e o funcionamento de múltiplos payloads de malware. Todas as ações foram realizadas exclusivamente para fins acadêmicos, sem execução dos arquivos gerados em ambiente Windows.

Inicialmente, foi realizado acesso ao Kali Linux via RDP e, no terminal, foi utilizado o comando `sudo -i` para obtenção de privilégios de superusuário. Em seguida, acessou-se o diretório do projeto por meio do comando `cd /curso/Brutal`, sendo confirmado seu conteúdo com `ls`.

O script principal `Brutal.sh` recebeu permissão de execução com `chmod +x Brutal.sh` e foi executado com `./Brutal.sh`, exibindo o menu interativo com diversas categorias de payloads.

Foi selecionada a opção **05 – Payload Prank for attack computer**, acessando o submenu de "brincadeiras" (hoaxes). Em seguida, foi escolhida novamente a opção **05 – Screen Rotation Prank**, resultando na criação do payload `Screen-Rotation-Pranks.ino`, conforme indicado pela mensagem de sucesso exibida no terminal (passo solicitado para print).

Posteriormente, utilizando o gerenciador de arquivos Thunar, foi acessado o diretório `/curso/Brutal/src/prank/`, onde foram visualizados os arquivos com extensão `.ino`. Esses arquivos correspondem a scripts que podem ser utilizados de forma maliciosa quando aplicados a dispositivos HID programáveis.

Atividade 2.5. Explorando como o Windows Defender detecta um Keylogger como programa malicioso

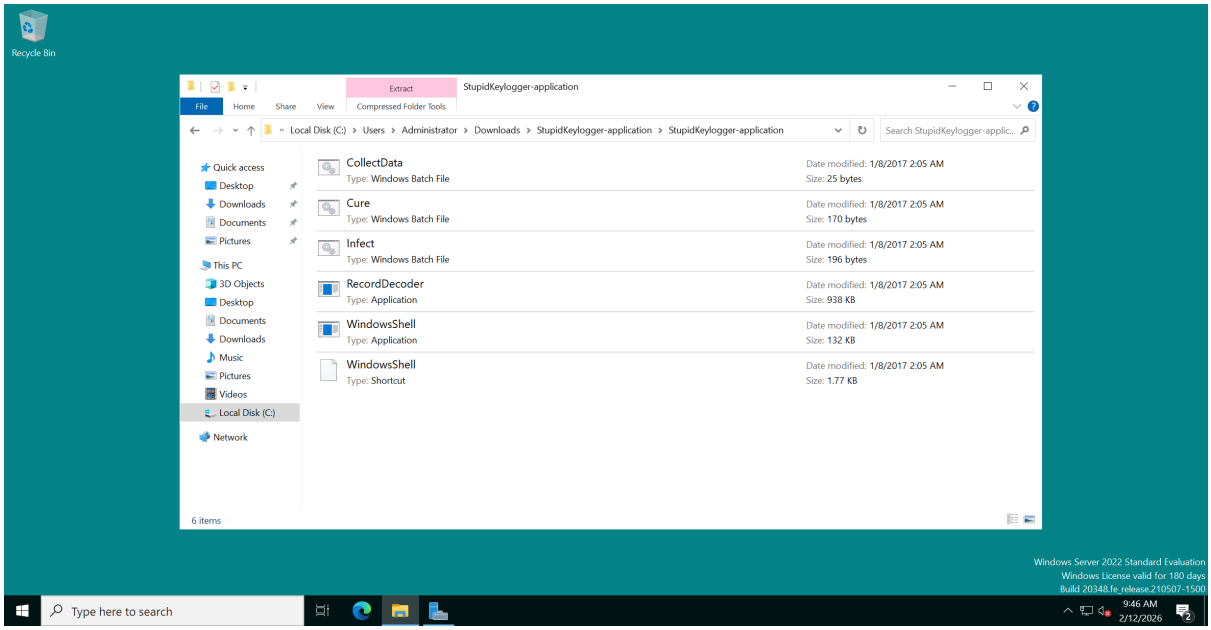

Fig. 5: Download do StupidKeylogger bloqueado pelo Microsoft Defender SmartScreen e visualização do arquivo na pasta Downloads

Sobre:

Nesta atividade, foi analisado o funcionamento dos mecanismos de proteção do Microsoft Defender SmartScreen ao tentar realizar o download de um Keylogger disponível publicamente no GitHub.

Após acessar o Windows Server 2022 (cliente) via RDP, foi utilizado o navegador Microsoft Edge para acessar o repositório `StupidKeylogger`, no qual é disponibilizado um exemplo de Keylogger para fins educacionais. A leitura da documentação permitiu compreender que o programa tem como finalidade capturar e registrar teclas digitadas no sistema operacional Windows.

Ao tentar baixar o arquivo compactado `StupidKeylogger-application.zip`, o Microsoft Defender SmartScreen bloqueou automaticamente o download, classificando-o como inseguro. Mesmo após selecionar as opções “Keep” e posteriormente “Keep anyway”, o sistema apresentou alertas adicionais indicando que o aplicativo era considerado potencialmente malicioso.

O arquivo foi então visualizado na pasta `Downloads` do Explorador de Arquivos, conforme solicitado no enunciado da atividade. Ressalta-se que o objetivo do procedimento foi exclusivamente observar o comportamento do mecanismo de defesa do Windows diante de um software potencialmente malicioso.