Hacking Ético Básico

Técnicas Iniciais de Teste de Vulnerabilidades

Página 1: Capa

Autor: Manus AI

Novembro de 2025

Página 2: Introdução ao Hacking Ético

O **Hacking Ético** é a prática de testar a segurança de um sistema de computador, rede ou aplicação, simulando ataques de *hackers* maliciosos, mas com a permissão explícita do proprietário do sistema. O objetivo não é causar dano, mas sim identificar vulnerabilidades para que possam ser corrigidas antes que sejam exploradas.

Tipos de Hackers

A ética é o que define a categoria do *hacker*:

Categoria	Descrição	Objetivo
White Hat	*Hackers* éticos. Trabalham legalmente para proteger sistemas.	Defesa e melhoria da segurança.
Black Hat	*Hackers* maliciosos. Invadem sistemas para ganho pessoal ou dano.	Exploração e causar dano.
Gray Hat	Atuam na zona cinzenta. Podem invadir sem permissão, mas com boas intenções.	Revelar vulnerabilidades publicamente.

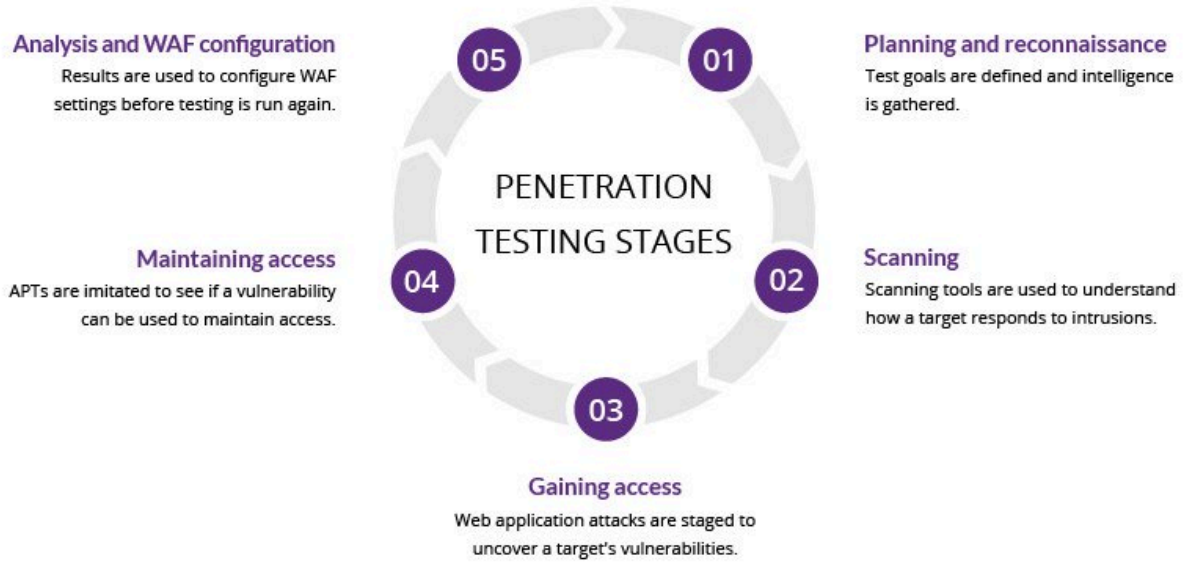
Objetivo: Defesa Proativa

O Hacking Ético é a essência da **Defesa Proativa**. Ao pensar como um atacante, o profissional de segurança consegue antecipar e mitigar riscos que, de outra forma, passariam despercebidos.

Página 3: O Ciclo do Hacking Ético (Pentest)

O Teste de Penetração (*Pentest*) é o processo formal que o *hacker* ético segue. Ele é dividido em fases que simulam o comportamento de um atacante real:

Fase	Descrição	Ferramentas Comuns
1. Reconhecimento	Coleta de informações sobre o alvo (passiva e ativa).	Google Dorks, WHOIS, Nmap.
2. Varredura	Identificação de portas abertas, serviços e sistemas operacionais.	Nmap, Nessus.
3. Ganhando Acesso	Exploração de vulnerabilidades para obter acesso ao sistema.	Metasploit.
4. Mantendo Acesso	Instalação de *backdoors* para acesso futuro.	Rootkits, Trojans.
5. Limpando Rastros	Remoção de logs e evidências da invasão.	Comandos de sistema.



Página 4: Fase 1: Reconhecimento Passivo

O **Reconhecimento** é a fase mais importante, pois a qualidade da informação coletada define o sucesso das fases seguintes. O Reconhecimento Passivo envolve a coleta de informações sem interagir diretamente com o alvo.

Ferramentas de Reconhecimento

- **Google Dorks:** Uso de operadores de busca avançados para encontrar informações sensíveis publicamente expostas (Ex: `filetype:xls site:target.com "senha"`).
- **WHOIS:** Consulta a bancos de dados públicos para obter informações de registro de domínio (contato, servidor DNS, data de expiração).
- **DNS Lookup:** Descoberta de subdomínios e endereços IP associados ao alvo.

Footprinting

É o processo de criar um "mapa" completo da pegada digital do alvo, incluindo endereços IP, domínios, subdomínios, e-mails de contato e tecnologias utilizadas.

Página 5: Fase 2: Varredura (Scanning)

A **Varredura** é a primeira interação ativa com o alvo. O objetivo é descobrir quais portas estão abertas e quais serviços estão rodando nelas.

Varredura de Portas (*Port Scanning*)

Cada porta aberta representa um serviço ativo e, potencialmente, uma vulnerabilidade.

Porta	Serviço Comum	Risco de Segurança
21	FTP (File Transfer Protocol)	Tráfego de credenciais e dados sem criptografia.
22	SSH (Secure Shell)	Se mal configurado, pode permitir ataque de força bruta.
80	HTTP (Hypertext Transfer Protocol)	Tráfego de dados em texto puro.
443	HTTPS (HTTP Secure)	Mais seguro, mas a configuração do certificado pode ser falha.

Nmap (Network Mapper)

O Nmap é a ferramenta padrão da indústria para varredura de rede.

```
# Varredura de portas TCP mais comuns (Top 1000)
nmap <IP_do_alvo>

# Varredura de portas com detecção de versão de serviço
nmap -sV <IP_do_alvo>

# Varredura de portas com detecção de sistema operacional
nmap -O <IP_do_alvo>
```

Página 6: Introdução à Análise de Tráfego de Rede

A **Análise de Tráfego de Rede** é a espinha dorsal da detecção de intrusão e da investigação forense. Ela envolve a captura e a inspeção de pacotes de dados que trafegam pela rede.

O que é um *Packet Sniffer*?

Um *Packet Sniffer* (farejador de pacotes) é um *software* ou *hardware* que pode interceptar e registrar o tráfego de dados que passa por uma rede digital.

A Importância dos Protocolos

Para analisar o tráfego, é fundamental entender o modelo TCP/IP:

Camada	Protocolos	Função
Aplicação	HTTP, DNS, FTP	Comunicação entre aplicações.
Transporte	TCP, UDP	Conexão e entrega de dados.
Internet	IP	Endereçamento e roteamento.
Acesso à Rede	Ethernet, Wi-Fi	Transmissão física.

O Wireshark como Ferramenta Essencial

O Wireshark é o analisador de protocolo de rede mais popular do mundo. Ele permite capturar e inspecionar o conteúdo de cada pacote, revelando informações cruciais como credenciais em texto puro, dados de sessão e anomalias de rede.

Página 7: Wireshark na Prática (Parte 1)

Interface do Wireshark

A interface é dividida em três painéis principais:

1. **Lista de Pacotes:** Exibe todos os pacotes capturados em ordem cronológica.
2. **Detalhes do Pacote:** Exibe o conteúdo do pacote em camadas (Ethernet, IP, TCP, HTTP, etc.).
3. **Bytes do Pacote:** Exibe o conteúdo do pacote em formato hexadecimal e ASCII.

Capturando Tráfego

Para capturar o tráfego, você deve selecionar a interface de rede correta (Wi-Fi, Ethernet) e, em muitos casos, executar o Wireshark em **Modo Promíscuo** para ver o tráfego que não é diretamente endereçado à sua máquina.

Filtros de Captura

Os filtros de captura são usados para limitar o volume de dados capturados, economizando espaço e tempo de análise. Eles são aplicados **antes** da captura.

Filtro de Captura	Descrição
`host 192.168.1.1`	Captura apenas o tráfego de/para o IP especificado.
`port 80`	Captura apenas o tráfego da porta 80 (HTTP).
`tcp or udp`	Captura apenas pacotes TCP ou UDP.

Página 8: Wireshark na Prática (Parte 2)

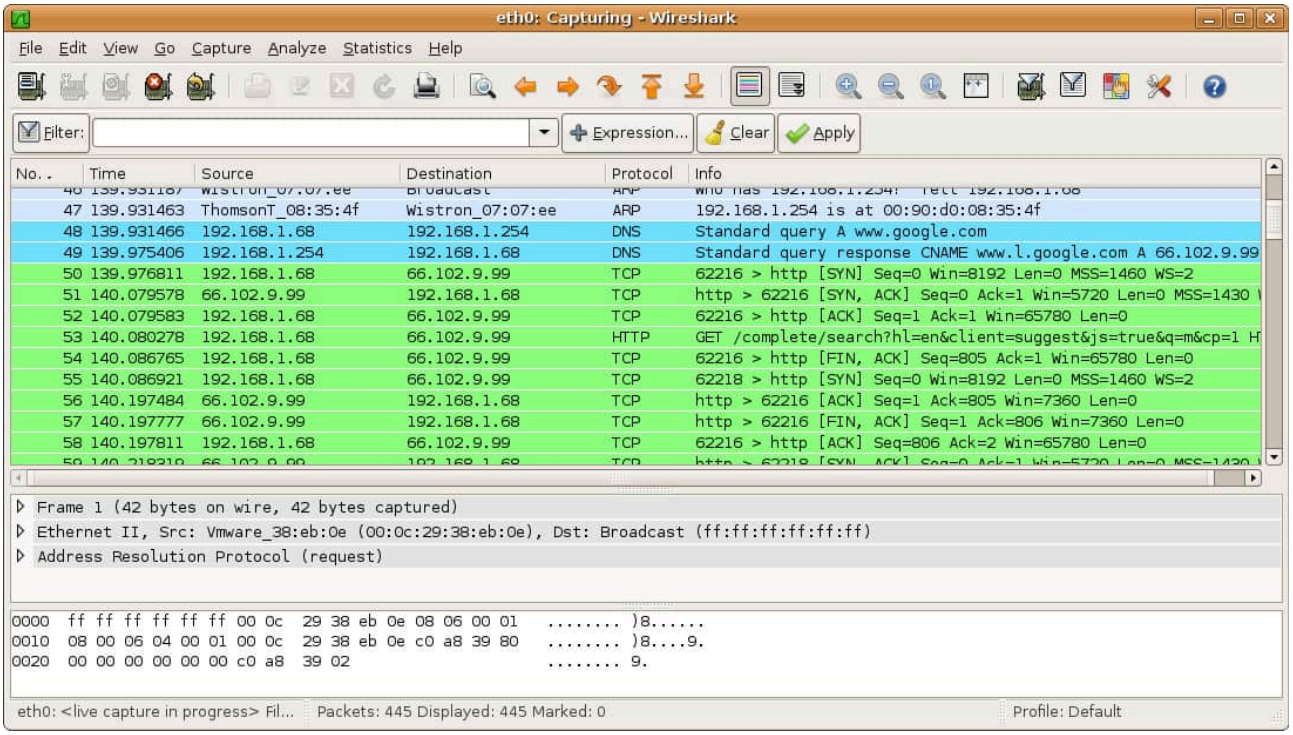
Filtros de Exibição

Os filtros de exibição são usados para refinar a visualização dos pacotes **após** a captura. Eles são mais poderosos e flexíveis que os filtros de captura.

Filtro de Exibição	Descrição
`http`	Exibe apenas pacotes HTTP.
`tcp.port == 80`	Exibe pacotes onde a porta TCP de origem OU destino é 80.
`ip.addr == 10.0.0.1`	Exibe pacotes onde o IP de origem OU destino é 10.0.0.1.
`http.request.method == "POST"`	Exibe apenas requisições HTTP do tipo POST.

Identificando Tráfego Não Criptografado

Um dos usos mais rápidos do Wireshark é identificar tráfego de protocolos legados (HTTP, Telnet, FTP) que transmitem dados em texto puro. Ao aplicar o filtro `http` ou `ftp` e inspecionar o painel de detalhes, é comum encontrar credenciais e dados sensíveis expostos.


eth0: Capturing - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply
No. Time Source Destination Protocol Info
47 139.931463 ThomsonT_08:35:4f Wistron_07:07:ee ARP 192.168.1.254 is at 00:90:d0:08:35:4f
48 139.931466 192.168.1.68 192.168.1.254 DNS Standard query A www.google.com
49 139.975406 192.168.1.254 192.168.1.68 DNS Standard query response CNAME www.l.google.com A 66.102.9.99
50 139.976811 192.168.1.68 66.102.9.99 TCP 62216 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
51 140.079578 66.102.9.99 192.168.1.68 TCP http > 62216 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430
52 140.079583 192.168.1.68 66.102.9.99 TCP 62216 > http [ACK] Seq=1 Ack=1 Win=65780 Len=0
53 140.080278 192.168.1.68 66.102.9.99 HTTP GET /complete/search?hl=en&client=suggest&js=true&q=m&cp=1
54 140.086765 192.168.1.68 66.102.9.99 TCP 62216 > http [FIN, ACK] Seq=805 Ack=1 Win=65780 Len=0
55 140.086921 192.168.1.68 66.102.9.99 TCP 62218 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
56 140.197484 66.102.9.99 192.168.1.68 TCP http > 62216 [ACK] Seq=1 Ack=805 Win=7360 Len=0
57 140.197777 66.102.9.99 192.168.1.68 TCP http > 62216 [FIN, ACK] Seq=1 Ack=806 Win=7360 Len=0
58 140.197811 192.168.1.68 66.102.9.99 TCP 62216 > http [ACK] Seq=806 Ack=2 Win=65780 Len=0
Frame 1 (42 bytes on wire, 42 bytes captured)
Ethernet II, Src: Vmware_38:eb:0e (00:0c:29:38:eb:0e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)
0000 ff ff ff ff ff ff 00 0c 29 38 eb 0e 08 06 00 01 ....)8......
0010 08 00 06 04 00 01 00 0c 29 38 eb 0e c0 a8 39 80 ....)8....9.
0020 00 00 00 00 00 00 c0 a8 39 02 .... 9.
eth0: <live capture in progress> Fil... Packets: 445 Displayed: 445 Marked: 0 Profile: Default

Página 9: Vulnerabilidades Comuns

O *hacker* ético busca vulnerabilidades que se enquadram em categorias bem conhecidas.

Vulnerabilidade	Descrição	Exemplo de Exploração
Injection	O atacante insere código malicioso (SQL, comandos de sistema) em um *input* da aplicação.	`SQL Injection` para extrair dados do banco de dados.
Broken Authentication	Falhas na gestão de sessões ou credenciais.	Senhas fracas, sessão que não expira, *brute force* em tela de login.
***Cross-Site Scripting* (XSS)**	Injeção de *scripts* maliciosos no lado do cliente (navegador) de um usuário.	Roubo de *cookies* de sessão.
Insecure Deserialization	Falha na desserialização de dados que permite a execução de código remoto.	Execução de comandos no servidor.

Página 10: Defesa Proativa

O conhecimento das técnicas de ataque é a base para a defesa.

Princípio do Menor Privilégio

Conceder a usuários, sistemas e aplicações apenas os privilégios mínimos necessários para realizar suas tarefas. Se um atacante comprometer uma conta, o dano será limitado.

Patch Management (Gerenciamento de Patches)

Manter todos os sistemas operacionais, *firmwares* e aplicações atualizados. A maioria dos ataques de sucesso explora vulnerabilidades conhecidas para as quais já existe uma correção (*patch*).

Uso de Firewalls e IDS/IPS

- **Firewall:** Controla o tráfego de rede com base em regras (portas, IPs).
- **IDS (Intrusion Detection System):** Monitora o tráfego em busca de padrões de ataque e gera alertas.
- **IPS (Intrusion Prevention System):** Monitora e ativamente bloqueia o tráfego malicioso.

Página 11: Boas Práticas e Ética

O Hacking Ético é regido por um código de conduta rigoroso.

A Importância do Contrato (*Scope*)

O *hacker* ético **NUNCA** deve iniciar um teste de penetração sem um contrato formal (*Rules of Engagement*) que defina:

1. **Escopo:** Quais sistemas, IPs e aplicações podem ser testados.
2. **Limites:** Quais técnicas são proibidas (Ex: *Denial of Service*).
3. **Horário:** Quando o teste pode ser realizado.

Princípio da Não Maleficência

O objetivo é encontrar a vulnerabilidade, não explorá-la ao máximo. O *hacker* ético deve parar imediatamente após comprovar a vulnerabilidade e não causar indisponibilidade ou perda de dados.

Reporte de Vulnerabilidades

O resultado final do *Pentest* é um relatório detalhado que inclui:

- **Resumo Executivo:** Para a gerência (risco de negócio).
- **Detalhes Técnicos:** Para a equipe de TI (como reproduzir e corrigir).
- **Recomendações:** Passos claros para a mitigação.

Página 12: Conclusão e Próximos Passos

O Hacking Ético é uma mentalidade de segurança que transforma o conhecimento de ataque em defesa.

Resumo das Ferramentas e Técnicas Essenciais:

Fase	Técnica	Ferramenta
Reconhecimento	*Footprinting*	Google Dorks, WHOIS
Varredura	*Port Scanning*	Nmap
Análise	*Packet Sniffing*	Wireshark
Defesa	*Patch Management*	Firewalls, IDS/IPS

Caminhos para a Certificação

Para aprofundamento e reconhecimento profissional:

1. **CEH (Certified Ethical Hacker):** Foco em ferramentas e metodologias de *pentest*.
2. **OSCP (Offensive Security Certified Professional):** Certificação prática e altamente respeitada, focada em exploração manual.

Lembre-se: O conhecimento é poder. Use-o para proteger.

Autor: Manus AI