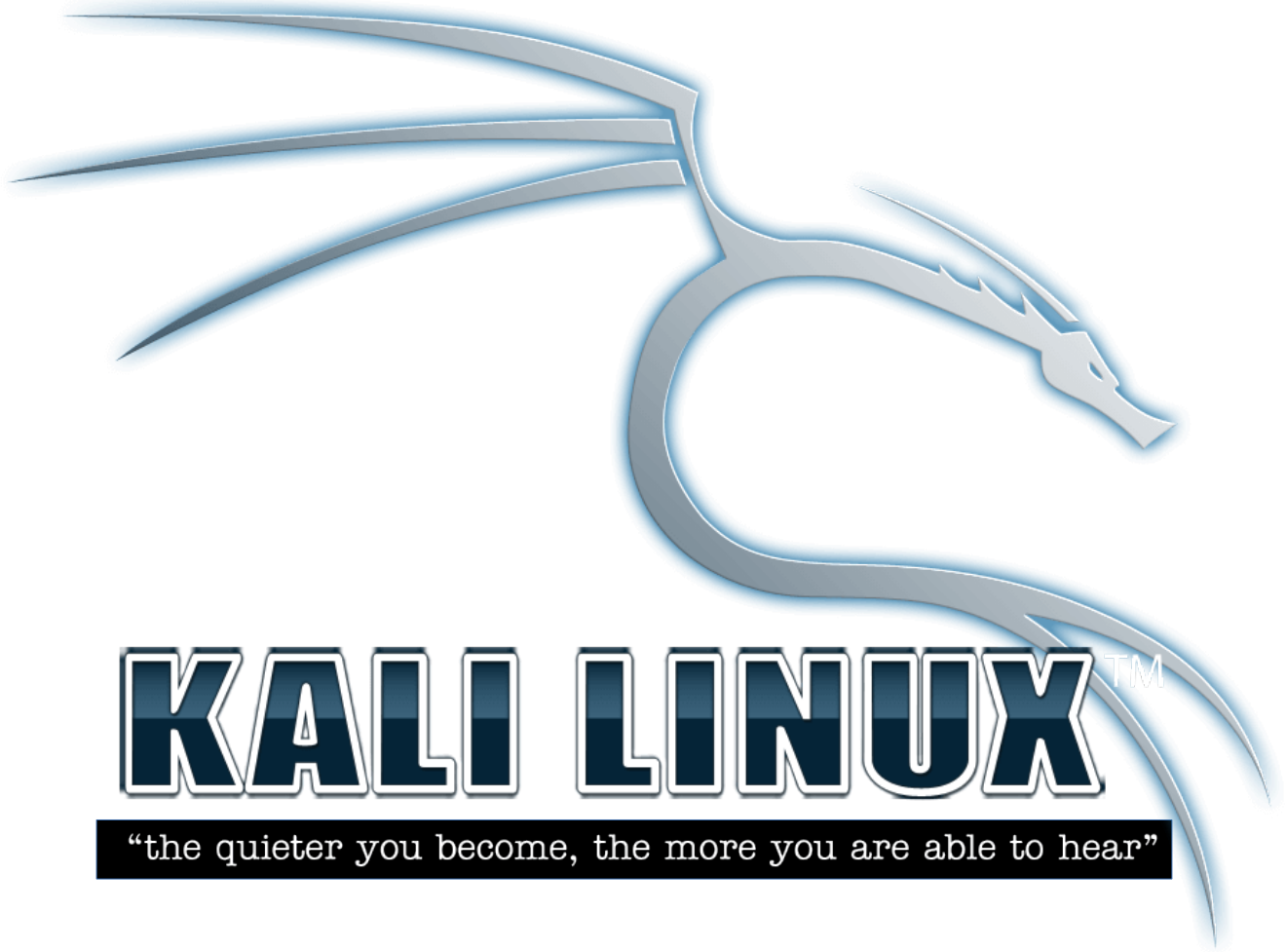
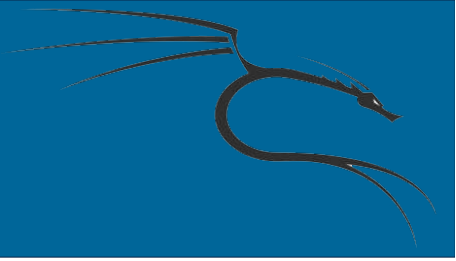
Pentest com Kali Linux





Instrutor:Vitor Mazuco

http://facebook.com/vitormazuco

Email:vitor.mazuco@gmail.com

WebSite:http://vmzsolutions.com.br

Nós, em um certo nível, assumimos que quando usamos uma conexão em HTTPS com SSL ou criptografia TLS, é protegido por qualquer atacante que se for capturado, só receberá uma série de números sem sentido.

Bem, isso pode não ser absolutamente verdadeiro; os servidores HTTPS precisam ser configurados corretamente para fornecer uma camada forte de criptografia e proteger os usuários dos ataques do tipo MiTM ou uma criptoanálise. Várias vulnerabilidades na implementação e no design do protocolo SSL foram descobertas ao longo dos anos; Assim, tornar obrigatório o teste de conexões seguras em qualquer teste de penetração de aplicações web é essencial.

Nesta aula, usaremos o SSLScan, uma ferramenta incluída no Kali Linux, para analisar a configuração (do ponto de vista do cliente) do servidor em termos de sua comunicação está mais segura. Vamos usar o OWASP como máquina vítima. Ele já vem com uma segurança em HTTPS.

O comando sslscan básico nos dará informações suficientes sobre o servidor:

sslscan 192.168.1.163

```
root@kali:~# sslscan 192.168.56.102
Version: -static
OpenSSL 1.0.1m-dev xx XXX xxxx

Testing SSL server 192.168.56.102 on port 443

  TLS renegotiation:
Secure session renegotiation supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.0 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.2 not vulnerable to heartbleed
```

A primeira parte da saída nos diz a configuração do servidor em termos de Configuração de segurança: renegociação, compressão e Heartbleed, que é um Vulnerabilidade encontrada em algumas implementações do TLS. Neste caso, tudo parece estar ok!

Nesta segunda parte, o SSLScan mostra os conjuntos de codificação aceitada pelo servidor, e como podemos ver, ele suporta o SSLv3 e algumas cifras como DES, que agora são considerados não segura; Eles são mostrados em cor vermelha, texto amarelo significa cifras de força média.

```
  Supported Server Cipher(s):
Accepted  SSLv3     256 bits  DHE-RSA-AES256-SHA
Accepted  SSLv3     256 bits  AES256-SHA
Accepted  SSLv3     128 bits  DHE-RSA-AES128-SHA
Accepted  SSLv3     128 bits  AES128-SHA
Accepted  SSLv3     128 bits  RC4-SHA
Accepted  SSLv3     128 bits  RC4-MD5
Accepted  SSLv3     112 bits  EDH-RSA-DES-CBC3-SHA
Accepted  SSLv3     112 bits  DES-CBC3-SHA
Accepted  TLSv1.0   256 bits  DHE-RSA-AES256-SHA
Accepted  TLSv1.0   256 bits  AES256-SHA
Accepted  TLSv1.0   128 bits  DHE-RSA-AES128-SHA
Accepted  TLSv1.0   128 bits  AES128-SHA
Accepted  TLSv1.0   128 bits  RC4-SHA
Accepted  TLSv1.0   128 bits  RC4-MD5
Accepted  TLSv1.0   112 bits  EDH-RSA-DES-CBC3-SHA
Accepted  TLSv1.0   112 bits  DES-CBC3-SHA
```

Por último, temos as cifras preferidas, as que o servidor vai tentar usar para comunicação, se o cliente aceitar; E finalmente, as informações sobre o Certificado que o servidor usa. Podemos ver que ele usa um algoritmo de força média para assinatura e uma chave RSA fraca. Diz-se que a chave é fraca porque tem 1024 bits de comprimento; Hoje em dia, as normas de segurança recomendam pelo menos 2048 bits.

E mais! O SSLScan não é a única ferramenta que pode pegar informações de criptografia de conexões SSL/TLS. Existe outra ferramenta incluída no Kali Linux chamada SSLyze que poderia ser usada como uma alternativa e às vezes pode dar resultados complementares aos nossos testes:

```
# sslyze --regular 192.168.1.163
```

As informações SSL/TLS também podem ser obtidas através de comandos OpenSSL:

```
# openssl s_client -connect www.example.com:443
```