# Coletando de informações em WebSites

Nesse artigo sera comentado sobre o programa DIRB e a sua utilizacao basica. Vejo poucos artigos escritos em português. (por isso a escolha do tema). Mas antes disso vamos nos atentar um pouco no assunto de Web Crawlers.

Quando se faz um Black Box sobre Web Sites algumas metodologias são seguidas (não será o foco tratá-las), como por exemplo, enumeração de servidores DNS's, serviços rodando, etc.

Um dos aspectos fundamentais quando se faz o teste de intrusão sobre Web Sites é conhecer bem como o site é feito e montado (servidor, quais são as páginas indexadas, etc.). Isso tudo é feito para que seja descoberto vulnerabilidades/erros.

Há várias ferramentas que automatizam o processo, tais como o w3af, nikto, arachni, etc. Porém vamos no deter em um detalhe mais simples: Web Crawlers. Mas…

## O que são Web Crawlers?

Web Crawlers são bots que sistematicamente navegam pelo Web Site, com o propósito de INDEXING (indexação de arquivos, páginas).

Basicamente o crawler irá "varrer" o web site por arquivos, páginas que estão indexados. Um ótimo exemplo disso… **Google** =D (digite, por exemplo, **inurl: www.site.com.br filetype: pdf**, será retornado arquivos PDFs que estão no seu site hoho =).

Para a varredura é passado ao programa crawler uma lista de palavras. Dessa forma, o programa irá checar se existe alguma URL, página, arquivo que possua o mesmo nome que está na lista de palavras informada. Caso exista, o crawler retornará um OK.

Não irei me prender a detalhes, mas a leitura de http://en.wikipedia.org/wiki/Web_crawler será bastante útil na definição e entendimento de crawlers.

## Um ótimo crawler: DIRB

O DIRB pode ser encontrado em http://sourceforge.net/projects/dirb (a distro KALI já o possui instalado hoho)

## Primeiro Teste: Um Teste Simples

Para um teste simples no **KALI** linux:

root@kali:~# cd /usr/share/dirb/

root@bt:/usr/share/dirb# dirb http://www.site.com.br

Será retornado, por exemplo, as páginas indexadas que possuem nomes que estão na wordlist (lista de palavras) padrão /usr/share/dirb/wordlist/common.txt.

Caso o seu site possua a URL http://www.site.com.br/Admin ou http://www.site.com.br/Downloads o DIRB retornará na tela:

—Scanning URL: http://www.site.com.br/ —

+ http://www.seusite.com/Downloads

(FOUND: 200 [Ok] – Size: 18470)

————————

DOWNLOADED: 1 – FOUND: 1

## Segundo Teste: Utilizando uma WordList customizada

Caso queira-se utilizar outra lista de palavras para se tentar retornar arquivos, páginas indexadas que não seja o common.txt (que é utilizado por padrão pelo DIRB) pode-se utilizar a seguinte sintaxe:

root@bt:/usr/share/dirb# dirb http://www.site.com.br caminho_da_sua_wordlist

O resultado será o mesmo para o primeiro teste. Caso a página exista será retornado:

— Scanning URL: http://www.site.com.br/ —

+ http://www.site.com.br/administracao

(FOUND: 200 [Ok] – Size: 18470)

————————

DOWNLOADED: 1 – FOUND: 1

**PS:** O dirb possui em seu diretório algumas *wordlists* muito boas. Elas podem sem encontradas em /usr/share/dirb/wordlists

## Terceiro Teste: Procurando arquivos PDF, RAR, FLV, etc.

Por fim, caso se queira procurar, por exemplo, um arquivo RAR a sintaxe a ser utilizada é essa:

root@bt:/usr/share/dirb# dirb http://www.site.com.br caminho_da_sua_wordlist (opcional, se for omitido será usado a wordlist padrão common.txt) -X .rar

O ponto deve ser colocado na opção **-X**, pois serão procurado arquivos.rar, se a opção **-X** for colocado sem o ponto será pesquisado arquivosrar

PS: Para testes em https a sintaxe é a mesma para o primeiro segundo e terceiro testes.

## Porque utilizar o DIRB?

Imaginem a seguinte situação:

Nosso o programador deixa seu site da seguinte forma:

http://www.site.com.br/admin/index.php

Do qual, você não consegue se logar. Pois quando tentar abrir o index.php, será pedido um usuário e senha (que supostamente você não tem!)

Porém… Fazendo uma coleta de dados com o DIRB obtemos a informação…

— Scanning URL: http://www.site.com.br/ —

+ http://seusite.com/admin/upload.php

(FOUND: 200 [Ok] – Size: 18470)

————————

DOWNLOADED: 1 – FOUND: 1

De tal forma que o index.php pedia autenticação, mas o upload.php não pede autenticação.

Temos dessa forma um formulario PHP para se fazer um upload de arquivos SEM nenhuma forma de autenticação.

E o Google não indexa esse tipo de formulário, apenas usando *crawlers* e um pouco de pesquisa obtém-se esse tipo de informação.

Grato,

Daniel Henrique Negri Moreno (a.k.a W1ckerMan)

Contato:

danielhnmoreno@gmail.com