## 

## Universidade Federal Rural de Pernambuco Sistemas de Informação Segurança e Auditoria de Sistemas Lhaíslla Eduarda Cavalcanti Rodrigues da Silva RECIFE-PE

2023

Buffer Overflow

## **Conceito**

Um buffer significa estouro, extrapolar o espaço que havia programado. Acontece quando um programa em execução tenta gravar dados fora do buffer de memória que não tem a função de armazenar esses dados.Buffer de memória é um espaço na memória RAM que serve para armazenamento temporário dos dados e serve para armazenar dados para entrada, saída e processamento. Quando ocorre um buffer overflow de memória os dados são gravados fora do buffer, e o programa em execução trava e retorna informações corrompidas, fica instável. As partes substituídas da memória são sobrescritas e arquivos importantes do aplicativo executado não ficam mais disponíveis ao programa. Estouros de buffer podem até executar outros programas ou comandos (maliciosos) e resultar em execução arbitrária de código.

Um buffer overflow funciona armazenando o conteúdo temporariamente, indicando onde os dados devem ser armazenados e copiando o endereço de destino onde se deve gravar os dados. Quando o programa é executado os dados da origem são copiados para o destino, sem haver uma verificação de limite, neste caso, quando o buffer de origem é maior que o buffer de destino, o buffer é excedido, ocasionando o estouro.

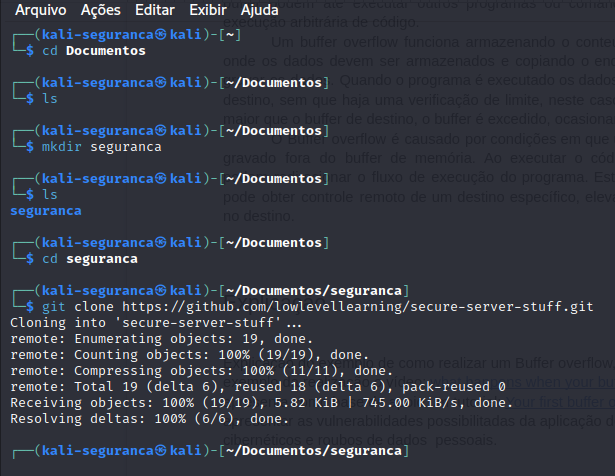
O Buffer overflow é causado por condições em que um programa em execução está gravado fora do buffer de memória. Ao executar o código(shell), programas maliciosos podem redirecionar o fluxo de execução do programa. Esta execução de código arbitrária pode obter controle remoto de um destino específico, elevar privilégios ou negar o serviço no destino.

## **Referências**

A explicação do exemplo de como realizar um Buffer overflow, utilizando como base de apoio o vídeo [what happens when your buffer are Too small](https://youtu.be/qpyRz5lkRjE) , que apresenta um tutorial: [Your first buffer overflow](https://medium.com/@lowlevellearning/your-first-buffer-overflow-b44e5ba5598a), com o intuito de abordar vulnerabilidades conforme a aplicação deste conceito, que pode ocasionarem ataques cibernéticos e roubos de dados pessoais.

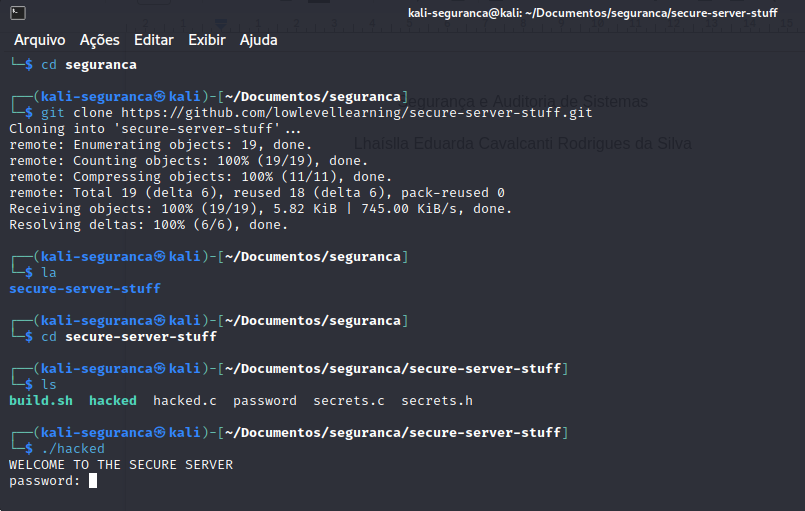
## **Aplicação**

Baixando repositório

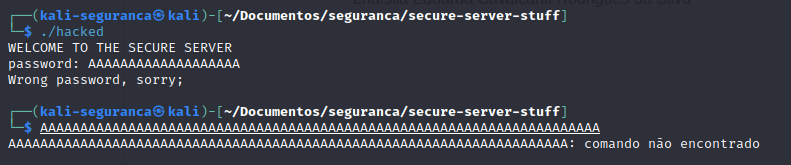


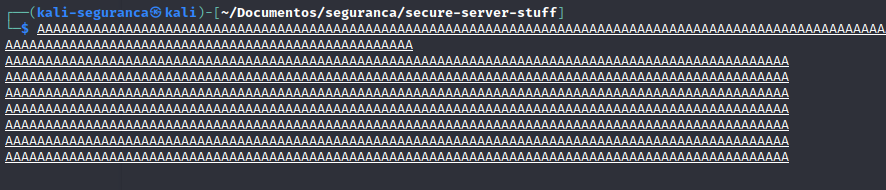
#### 

#### **Acessando o executável hacked**



#### **Executando O arquivo servidor** e verificando a funcionalidade das senhas, com senhas padrão e um maior tamanho de caracteres.



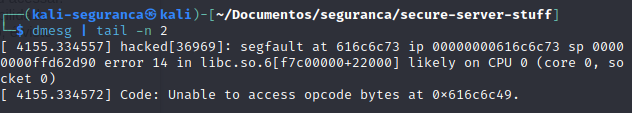


#### **Estouro de memória:** Em situações como está, onde ocorre o travamento, o atacante pode se aproveitar do estouro do buffer para fornecer mais dados do que o servidor consegue suportar.

#### 

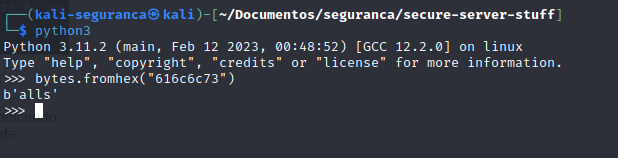
#### **Falha de segmentação:** O programa tenta acessar a memória que não conseguiu acessar.

Este fator é um indicador de preocupação para o programador, pois indica vulnerabilidade de arquitetura do programa, já dá perspectiva do hacker, quando isto é identificado, é visto como uma oportunidade de ataque.

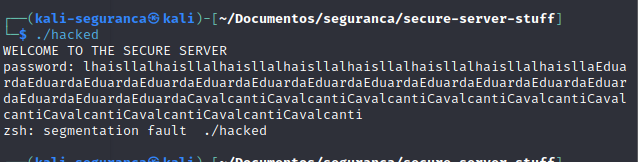


#### 

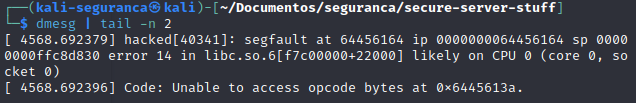
#### **Verificação de falha do arquivo:** É retornado o arquivo hexadecimal da string, indicando que a strins estourou no endereço de retorno e a CPU tentou executá-la.

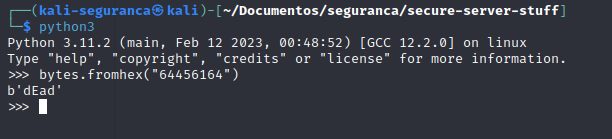


#### **Controle de falhas:** Com a inclusão de caracteres adicionais é possível verificar onde o controle de caracteres começa.



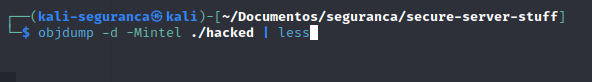
#### **Identificando qual a parte da string causou a falha:** É retornado um novo endereço, 64456164, que é correspondente aos caracteres ‘dEad’.





Com isto é possível determinar onde o estouro assume o controle do ponteiro de retorno. Diante disto, se faz necessário definir um endereço no buffer, para retornar a função de depuração, com isto, o endereço de strings erá colocado no lugar da string ‘dEad’.

#### **Determinar o endereço de depuração:**

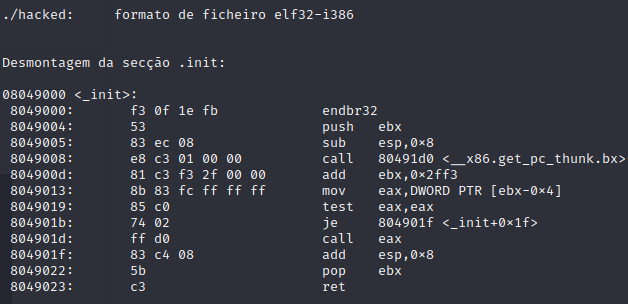
****

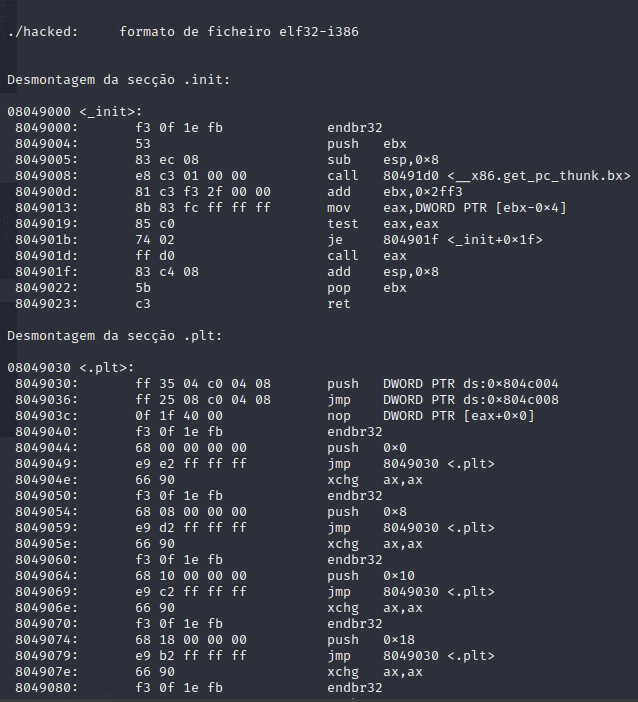
#### **Identificando o endereço de depuração:**

É utilizando o comando

***objdum -d Mintel ./haked | less***

resultando no seguinte endereço de depuração 804901f

****

****

#### **Criando uma nova entrada de estouro do buffer:** Ao inserir o endereço de depuração onde estava a string, há a possibilidade de controlar o programa e voltar a depurar sem a necessidade da senha. Utilizando python, é criado uma entrada de estouro do buffer, colocando o endereço de depuração no lugar correto na pilha. Substituindo o “dEad” da string pelo endereço de depuração, possibilitando assumir o controle do programa.

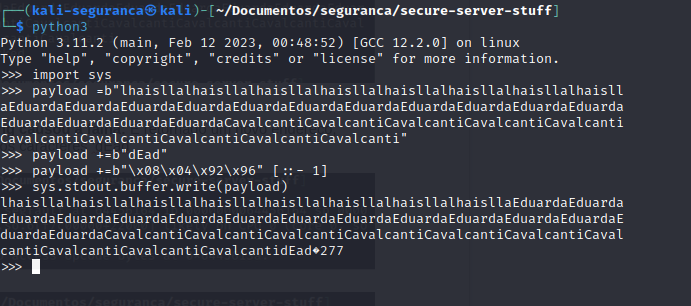
import sys

payload=b"lhaisllalhaisllalhaisllalhaisllalhaisllalhaisllalhaisllalhaisllaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaEduardaCavalcantiCavalcantiCavalcantiCavalcantiCavalcantiCavalcantiCavalcantiCavalcantiCavalcantiCavalcanti"

payload +=b"dEad"

payload +=b"\x08\x04\x92\x96" [::- 1]

sys.stdout.buffer.write(payload)



#### 

#### **Resultado:** Ao colocar o endereço de depuração no lugar correto na pilha, substituindo o “dEad” da string pelo endereço de depuração, feito isto se assume o controle do buffer.

