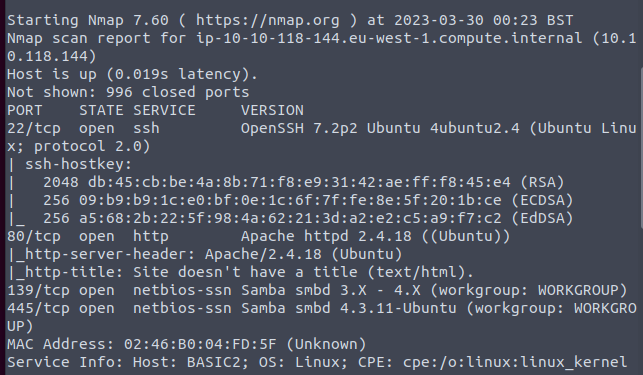
Passo 1 - reconhecimento e enumeração

Fizemos o reconhecimento da rede utilizando o: **nmap -sC -sV ip**

O -sV investiga todas as portas abertas que encontra para determinar se podemos obter as informações de serviço/versão.

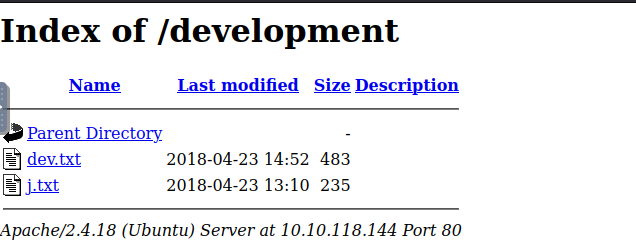
O sinalizador -sC também executa scripts em portas abertas para determinar se há vulnerabilidades externas/comuns que podemos usar imediatamente



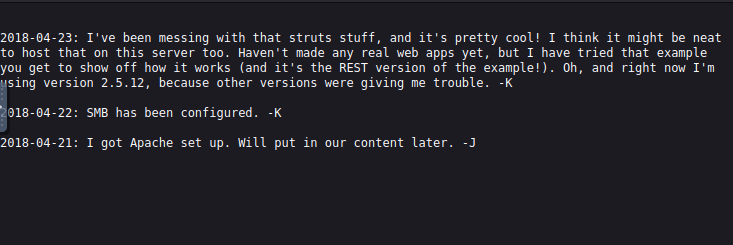
Como tem a porta 80 aberta, sabemos que há um site, então fizemos um **dirb http://ip**, para enumerar os diretórios existentes:

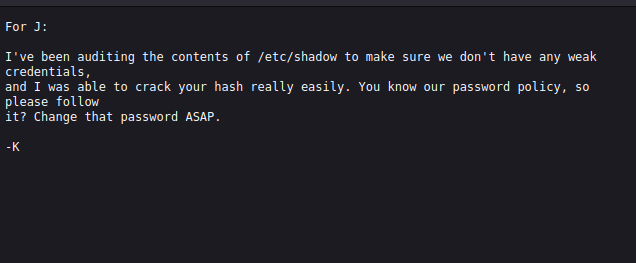


Encontramos 3 diretórios, ao analisá-los descobrimos que o development havia informações que nitidamente não deveriam estar ali:

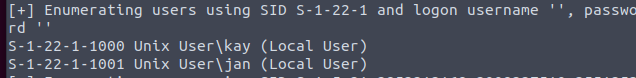


Havia troca de mensagens entre o -J e o -K

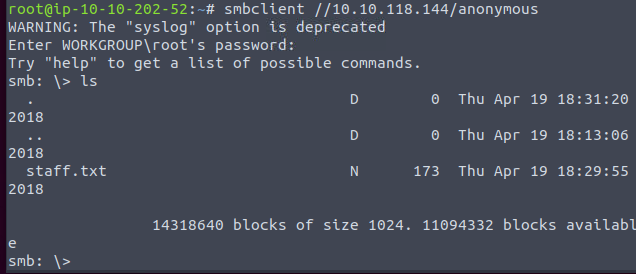




Como não havia nenhum usuário ou senha, resolvemos fazer uma enumeração de usuários no smb utilizando o **enum4linux ip**(é uma ferramenta para enumerar usuário windows e do smb/samba), e com isso tivemos o retorno de 2 usuários, o Jay e o Kay.

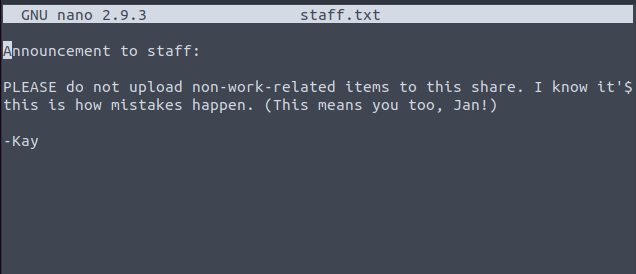


Também testamos o usuário anonymous do smb: **smbclient //ip/anonymous** para testarmos se conseguimos acessar com o usuário anonymous e sem senha. E deu certo:



baixamos o usuário staff.txt com o comando **get staff.txt**

E ao abrirmos o arquivo (para abrir o arquivo, você terá que sair do smb com o comando exit, e dará um ls na sua máquina normal, lá irá aparecer algumas pastas como Desktop, Download, e também terá o arquivo baixado staff.txt, para abrir é só dar **nano staff.txt**) vimos outra troca de mensagem:



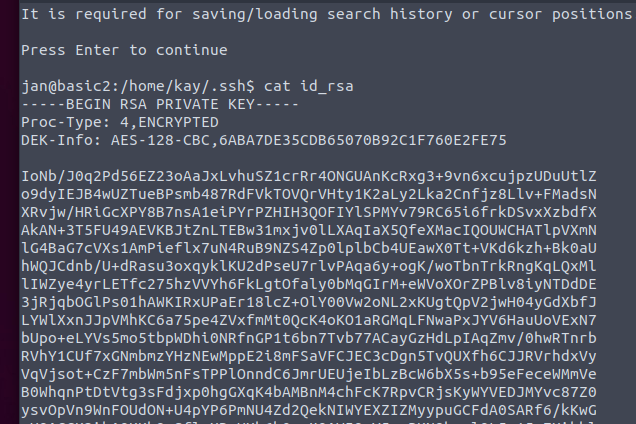
Ao ler as mensagens, é possível verificar que o Jan não segue as políticas de senha, ou seja, a senha dele provavelmente é fraca, com isso iremos tentar fazer uma conexão ssh, mas antes temos que descobrir a senha do Jan, iremos utilizar o seguinte comando: **hydra -l jan -P /usr/share/wordlists/rockyou.txt ip ssh** (hydra é usada para força bruta de nomes de usuário e senhas para diferentes serviços, como FTP, ssh, telnet, MS SQL, etc. A força bruta pode ser usada para tentar diferentes nomes de usuário e senhas contra um alvo para identificar as credenciais corretas.), como já sabemos o nome do usuário, colocamos apenas uma lista de senhas para a ferramenta, no caso a lista rockyou. E descobrimos a senha:

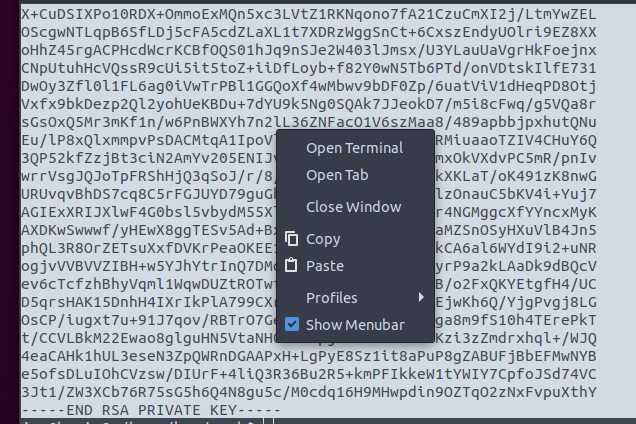


Agora para acessar o ssh, usamos o comando: **ssh jan@ip,** yes, e colocamos a senha, e conseguimos logar:



Ao dar o /home, utilizando o comando ls -la, verificamos que havia uma pasta chamanda kay, e sela outra pasta chamada .ssh (para chegar nela é preciso digitar: cd /home/kay/.ssh), e vimos que havia as chaves de conexão automática do ssh (id\_rsa) Copei ela do —---- Begin, até o final do arquivo (para abrir o arquivo dei **um cat id\_rsa**)





Sai do Jay e voltei para a minha máquina normal, onde criei o arquivo id\_rsa

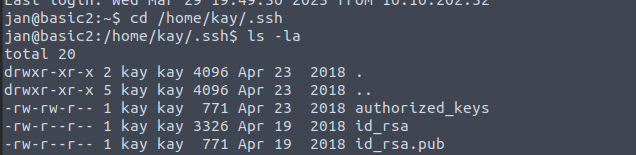


Para decriptar a chave, transformei ela em um arquivo hash que a ferramenta john consiga decifrar, utilizando o seguinte comando: **python /opt/john/ssh2john.py id\_rsa > hash.txt,**

Com isso utilizando o comando: **john –wordlist=”/usr/share/wordlists/rockyou.txt”** para ele tentar decifrar ( no caso, tem a lista rockyou, onde há milhares de senha possíveis que já foram vazadas, ele faz uma comparação de hash, se a hash for igual a hash de alguma senha da lista ele consegue mostrar pra gente qual a senha). Como vemos abaixo descobrimos a senha beeswax



Após isso nos conectamos no jay novamente, e fizemos a conexão ssh do usuário kay:



utilizando o comando: **ssh -i id\_rsa kay@ip, yes,** colocamos a senha beeswax, e conseguimos visualizar o arquivo escondido:

