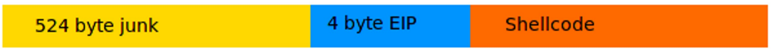
Vulnhub Brainpan

Friday, 29 July 2022 8:08 PM

Brainpan 1 - vulnhub
YouTube video walkthrough
Explained steps walkthrough
2nd Explained steps walkthrough (Corelan Resource link)

Things learnt

- Use netdiscover -r networkip/mask to see active hosts on a network, this gives you the vulnerable machines IP
- Do a simple nmap -p- scan to see all active ports. This could save time as opposed to adding all flags at once and being overwhelmed with information to read
- Check the service / version of ports and the operating system of the host with nmap to see if there are any vulnerabilities associated with them
- Dirbuster is a tool that bruteforces the directories of a website
- The file command returns filetype information
- The strings command displays the text within a binary / data file. This is useful for .exe files
- Network server applications (.exe files) use functions socket, bind, send, recv, listen and accept
- The strcpy function is vulnerable to a buffer overflow attack
- The wine command lets you run windows executables on windows
- A fuzzer? program connects to a network server and sends a huge payload to get it to crash (buffer overflow attack)
- What does a debugger like (immunity or local win) do? They allow inspection of code at a more granular level as each instruction can be ran at the pen tester's / attacker's pace rather than the processor's pace.
- The buffer overflow in this attack used does the following:
    - Sends a payload containing enough bytes to overwrite the EIP with the value of the "JMP ESP" instruction which directs the execution flow to the shellcode located in the stack / ESP.
    - 

- The shellcode in this case are instructions to set up a reverse shell.
- Msfvenom, msfpayload and msfencode are used to convert commands on linux and windows into instructions.
- When connected to the target machine:
    - Check which user you're logged in as with whoami and id
    - Check the kernel running and its version with uname -a
    - Look for SUID/SGID binaries and file perm misconfigs with a find command OR search files owned by each user
    - Main priority is to find a way to elevate privileges.

Summary:
The Brainpan vm had a web server running on one of the two ports which used strcpy to copy the password input. Strcpy is susceptible to buffer overflow attacks, which means that the EIP register can be overwritten when the program is in execution. A carefully designed payload that is sent to the server can overwrite the heap, the EIP register and the stack, with the EIP containing the JMP ESP instruction, which is directing the execution flow to the stack, which contains the malicious shellcode. In order to gain control of the VM the malicious shellcode was essentially instructing the host to reverse shell with the destination IP as the attackers IP and the desired destination port. Once the attacking machine had a reverse shell to the VM, in order to elevate privileges there was a

program owned by another user that could be executed as if you were the owner of the file (SUID misconfiguration). The program appeared to be using the strcpy function, which was susceptible to buffer overflow attacks. The EIP register was replaced with a JMP eax, which jumps to the address containing the shellcode to return a command line interpreter. Which allows puck to now access the server as anansi. The anansi user had access to a file that could be executed as SUDO without password. This file also could be rwx (read, overwritten) by anansi allowing the contents of the file to be replaced with /bin/bash, and executing that, giving you access as SUDO.