|  |  |
| --- | --- |
| Describe | Vulnerability |
| Service | HTTP |
| Port | 80 |
| Version | 2.2.22 |
| Severity | hard |
| Cvc I’d |  |
| Cvss score | 7/10 |
| Remidiations | 1. Exposure of Sensitive Information via robots.txt:  Vulnerability: The robots.txt file discloses multiple directories, some containing sensitive information such as potential passwords.  Remediation:  Restrict Access to Sensitive Directories: Configure the web server to prevent unauthorized access to directories listed in robots.txt.  Limit robots.txt Entries: Only include non-sensitive paths in robots.txt and avoid referencing directories that could expose critical information.  2. Storage of Plaintext Passwords in Web Directories:  Vulnerability: Passwords are stored in plaintext within web-accessible directories, making them easily retrievable.  Remediation:  Remove Plaintext Passwords: Ensure that sensitive information is not stored in web-accessible locations.  Use Environment Variables: Store sensitive data, such as passwords, in environment variables or secure vaults, not within the web root.  3. Use of Default or Weak Credentials:  Vulnerability: The application utilizes default or easily guessable credentials, facilitating unauthorized access.  Remediation:  Enforce Strong Password Policies: Require users to create complex passwords that meet defined security criteria.  Disable Default Accounts: Remove or disable default accounts and ensure all active accounts have unique, strong credentials.  4. Unrestricted File Upload Leading to Remote Code Execution:  Vulnerability: The application allows users to upload files without proper validation, enabling the upload of malicious scripts.  Remediation:  Implement File Type Validation: Restrict uploads to only necessary file types and validate file headers to confirm their content.  Rename Uploaded Files: Assign unique names to uploaded files to prevent overwriting and avoid execution of malicious scripts.  Store Files Outside Web Root: Save uploaded files in directories not accessible via the web to prevent direct execution.  5. Insecure Use of Sudo Permissions:  Vulnerability: Certain binaries, such as Perl, are allowed to run with elevated privileges without proper restrictions, enabling privilege escalation.  Remediation:  Restrict Sudo Access: Limit sudo permissions to only those commands necessary for specific users, and avoid granting unrestricted access to interpreters like Perl. |
| POC | Step 1  First we find IP and do netdiscover for Dina IP  (Netdiscover -r 192.168.1.106)  A screenshot of a computer program  Description automatically generated  After that we do,Nmap for checking open ports  (nmap -sT 192.168.1.105)  A computer screen with white text  Description automatically generated  Step 2  After doing Nmap, now we do ‘dirb’  (dirb :- <http://192.168.1.105/> )  A screenshot of a computer  Description automatically generated  Now we get “robot.txt” lets see what inside in robot.txt  A screenshot of a computer  Description automatically generated  We get lot of directories Now let’s check one-by-one all of them!!  A screenshot of a computer  Description automatically generated  After checking all of directory we get something in “nothing” directory  Step 3  Now I do “curl” for looking source page  A computer screen with text  Description automatically generated  I get secret pass, now I save this now for future  Now I research in dirb but now I get something imp lets search that!!  ( <http://192.168.1.105/nothing/> )  A screenshot of a computer  Description automatically generated  After open I found “backup.zip” file  Step 4  Now I download this file with the help of ‘wget’  ( wget <http://192.168.1.105/secure/backup.zip> )  A black background with a blue and white design  Description automatically generated with medium confidence  Lets read this file with “cat command”  A computer screen shot of a computer code  Description automatically generated  Step 5  This is file is now encrypted form. So now I used the ‘zip2john’ tool to extract the password hash first.  Then I used the john tool along with the passwords  that we gathering from the webpage to crack the hash. This is how I did it  A screen shot of a computer program  Description automatically generated  A computer screen with white text  Description automatically generated  Step 6  After crack this, now I use ‘7z x backup.zip’ and put password ‘freedom’ for looking sensitive information from that  A computer screen shot of a computer  Description automatically generated  After this I get ‘backup-cred.mp3’ file I using ‘cat’ command for looking inside in that  A black background with white text  Description automatically generated  As you see I get URL “ /secretSMSgatwaylogin ”  Step 7  After opening URL I get login page in playsms  A screenshot of a computer  Description automatically generated  We know uname - touhid  For password I use that file I get from ‘robot.txt’  After using that file, I get password ‘Diana’  A screenshot of a computer  Description automatically generated  As you see I login successfully  Lets try to find more information in that  Step 8  I open ‘msfconsole’ for getting access of root  After that I use cmd ‘search playsms’    After use I ‘#0’  A screenshot of a computer  Description automatically generated  In that I fill  (PASSWORD,RHOST,TARGETURI,USERNAME,LHOST)  A computer screen shot of a program  Description automatically generated  After that giving cmd to meterpreter “shell”  Step 9  After that open new terminal and use ‘nc’ and then use sudo  [nc -nlvp (port no)]  [sudo /usr/bin/perl -e ‘exec (“/bin/bash”)’; ]  Use (cd /root)  ‘Ls’ then we get “flag.txt”  Then ‘cat flag.txt’  And boom we get root flag of dina !!  A computer screen shot of a computer program  Description automatically generated |
| reference | https://medium.com/@anushibin007/dina-1-0-1-vulnhub-walkthrough-4160122ad0bc |