Proposal for GSoC 2021

Improving the Post Exploitation API to make it more consistent across different sessions

Abstract

Metasploit is the world's most popular and most used penetration testing framework. It’s an open source project known for developing and executing exploit code against target machines. It has a very large number of modules which performs various tasks from information gathering to cleaning up the tracks during a penetration test. The goal of this project is to improve the Post Exploitation API so that it can work more consistently and smoothly across different sessions.

Objective

The objective of the project is to improve the post exploitation API which is the part of metasploit framework with which the user interacts after getting the shell from target. However, at some places it is not much smooth and could be improved. One such area would be to implement the same file system API to shell sessions that meterpreter sessions use. The other primary objective of the project is that everything that is added either as a feature or as a solution to some bug should make the tool better for the end users or module developers.

Skills and languages

- Major Pentesting tools
- Git
- Ruby
- Linux Privilege Escalation
- Windows Privilege Escalation
- Computer Networks
- Network Pentesting
- Web Pentesting
- Binary Exploitation
- PHP
- Python
- C
- SQL
- OSINT

Timeline

Pre GSoC (Before May 17th)

- Read the wikis for development.
- Went through many parts of the codebase like protocol libraries, console related libraries, scanners etc.
- Did some small contributions by solving some issues and raising them when encountered.
- Had some discussions with mentors and core members about the codebase and future enhancements.
- Looked at some libraries which are present outside the metasploit-framework repository like rex-text.

Community Bonding Period (May 17th - June 7th)

- Introduce myself and this project on Slack#GSoC
- Remain in constant touch with my mentors and set up the environment.
- Discuss about the implementation plan with the mentors.
- Go through the code and try to add features and find issues in the existing codebase to understand it better.
- Setup a new Document for writing activities I do for the entire Summer.

Official Coding Period (June 7th - August 16th)

June 7th - June 22nd

- Go through the core Post Exploitation Libraries of various operating systems and the common ones.
- Understand the working of meterpreter.
- Go through the rex-post library.
- Understand the meterpreter filesystem API.

June 22nd - July 7th

- Form a plan to use meterpreter filesystem API for shell sessions.
- Discuss the plan with mentors.
- Implement meterpreter filesystem API in Shell sessions.
- Deliver the documentation and tests.

July 7th - July 12th

- Form a plan to add API to meterpreter to run simple commands through cmd_exec.
- Discuss the plan with mentors.

July 12th - July 16th

- Evaluations

July 16th - July 22nd

- Implement cmd_exec API for simple commands of meterpreter.
- Deliver the documentation and tests.

July 22nd - August 2nd

- Plan for adding localization support for Post Exploitation API.
- Discuss the plan with mentors.
- Implement the plan.
- Deliver the documentation and tests.

August 2nd - August 16th

- Buffer Time for unexpected delay or emergency.
- Fix bugs and update documentation.
- Pull request for code review and merge.

Post GSoC (After August 16th)

- Continue as an active contributor and remain in touch with community members and mentors.
- Document the parts of codebase which I come across and find undocumented.

Inspiration for Organization

There were many reasons why I choose Metasploit-Framework :

- Super Active Community
- Welcoming nature of community
- Fabulous Documentation
- Presence of many experts of various fields
- Ruby Codebase
- Best Pentesting Framework

Biography

I'm currently enrolled as a student in the Department of Computer Science, Vidyavardhaka College of Engineering. I'm also a Cybersecurity student and Open Source enthusiast. I love playing CTFs and cyber war games on platforms like HackTheBox, Rootme and Vulnhub. I like to understand how things work and how we can break it. I like reading code in various languages. I've also conducted some workshops on Pentesting, Linux and Git for Open Source Community, VVCE, which is a student run community of FOSS enthusiasts.
I've attended some conferences and seminars like OSI(Open Source India) and ACM winter School on Cyber Security. I like writing scripts for automating stuff. I am interested in all tracks of cyber security but the track which interests me the most is system internals.

I’m also a moderator of Open Source Community, OSL and attend many open source meetups every year. It’s also a place where I work before and after college, and learn and discuss new concepts and technologies with other members. We also organize and participate in coding contests and monthly code sprints.

I’m am playing CTFs from 1 year on different platforms like HTB, THM, Pico CTF and some others. I have written one exploit for CVE-2019-17240 which bypasses the password limit protection and have also build a small terminal based chat system with the help of ruby, multithreading and socket Programming
Apart from all that my hobbies also include watching football and playing PC games.

GSoC Participation

This is the first time I’m applying in Google Summer of Code. I did not submit a proposal to any other organization.

Personal Details

Name: Gaurav Purswani
College: Vidyavardhaka College of Engineering, Mysuru, India
Email: gauravpurswani1234@gmail.com
Country: India
Github/Twitter/Telegram: pingport80
Timezone: IST (GMT + 5:30)
Languages: English, Hindi, Sindhi