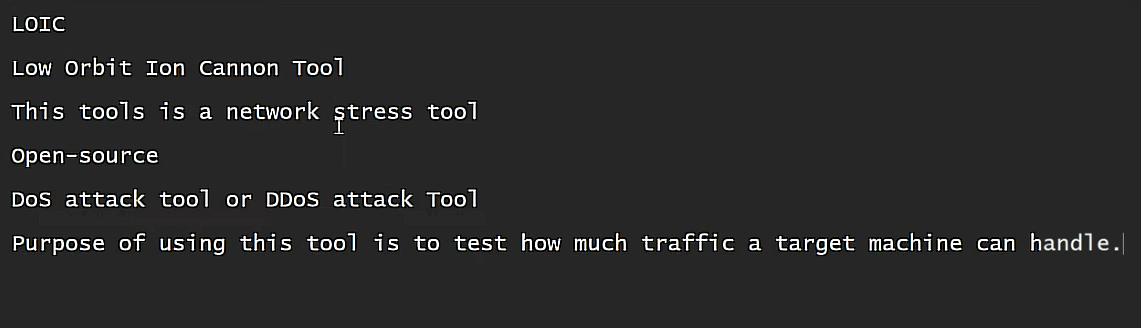
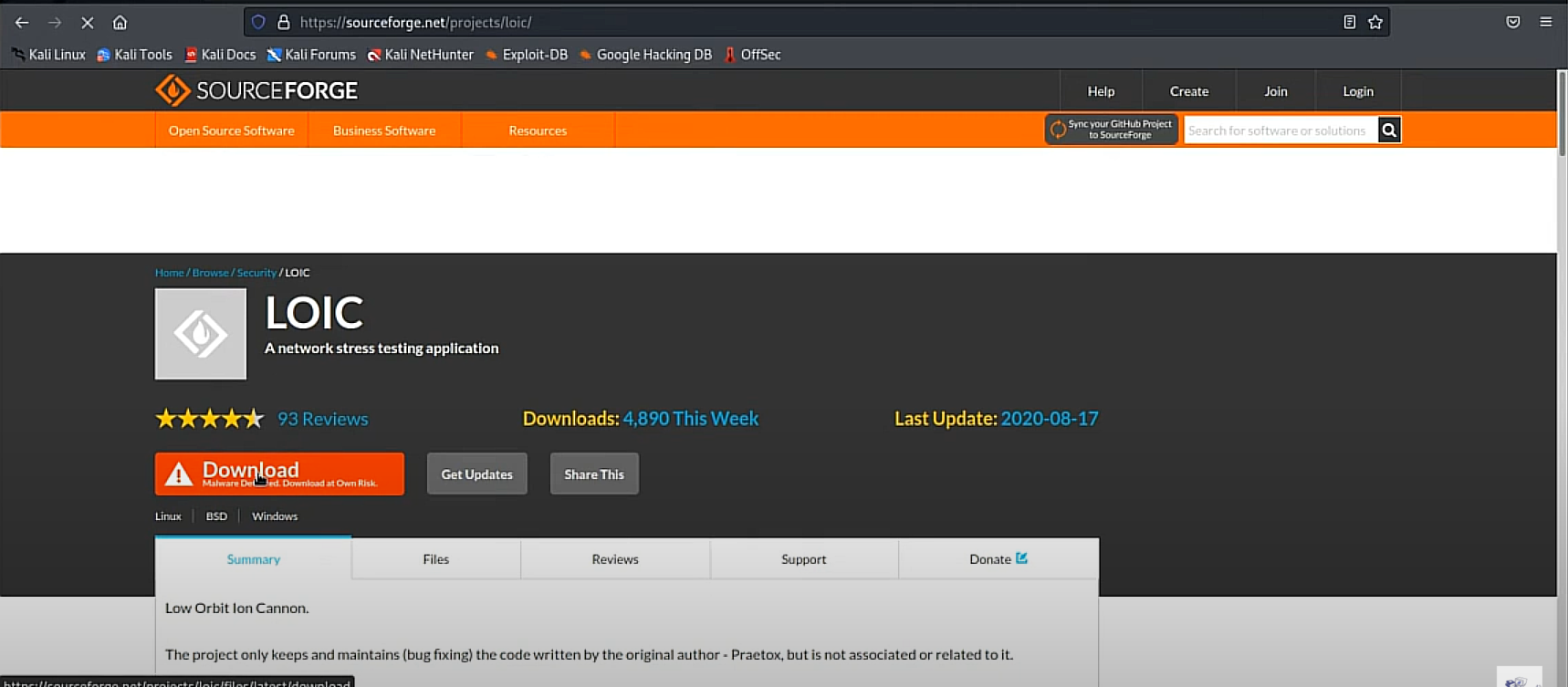
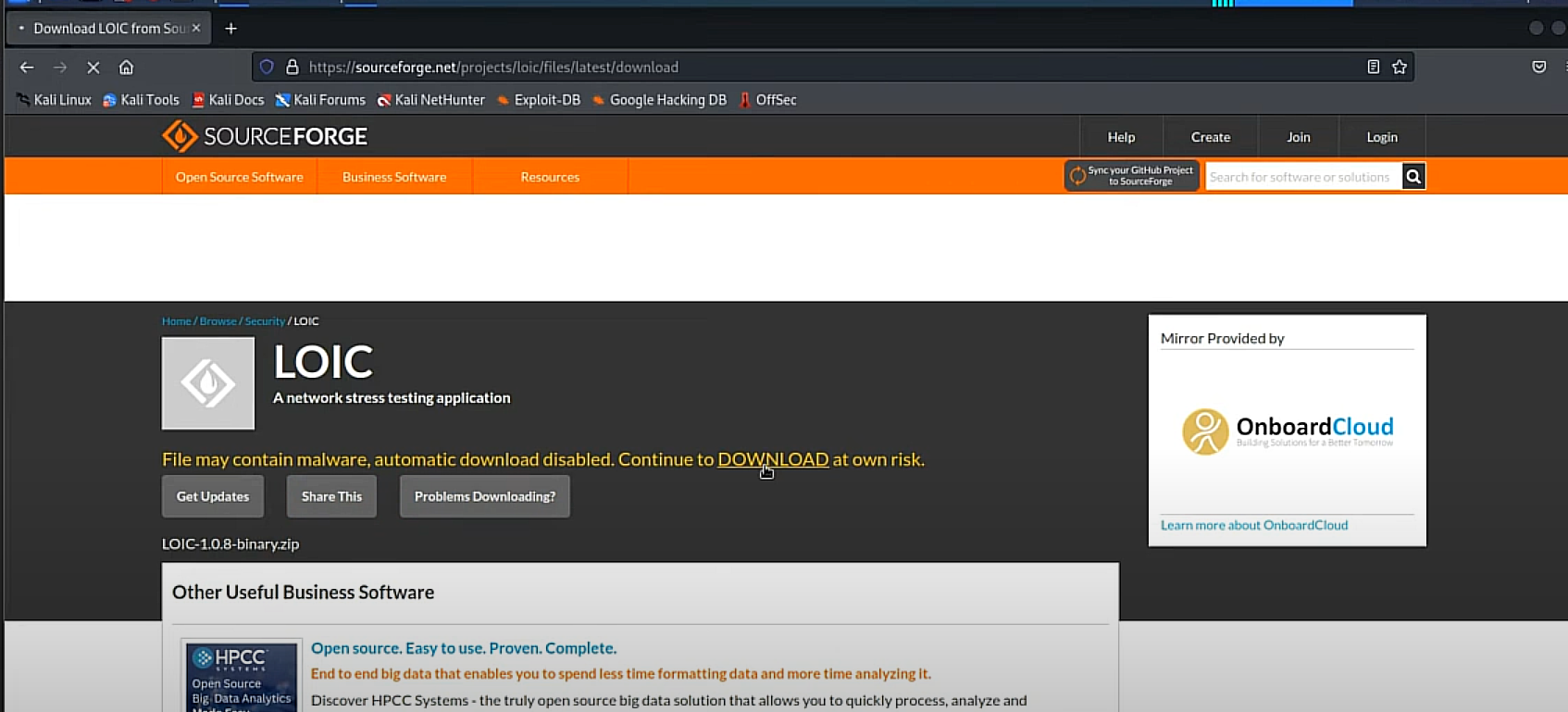
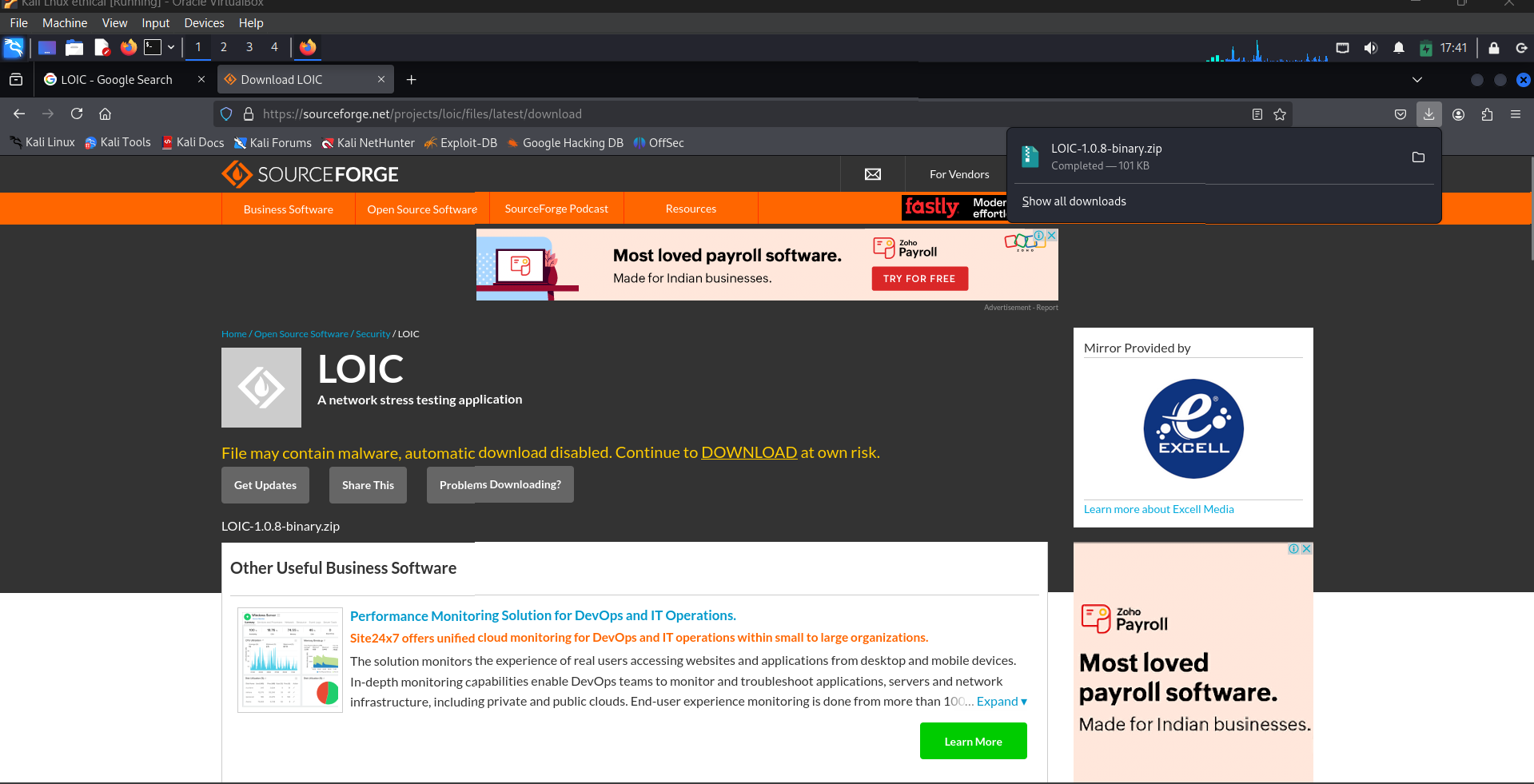
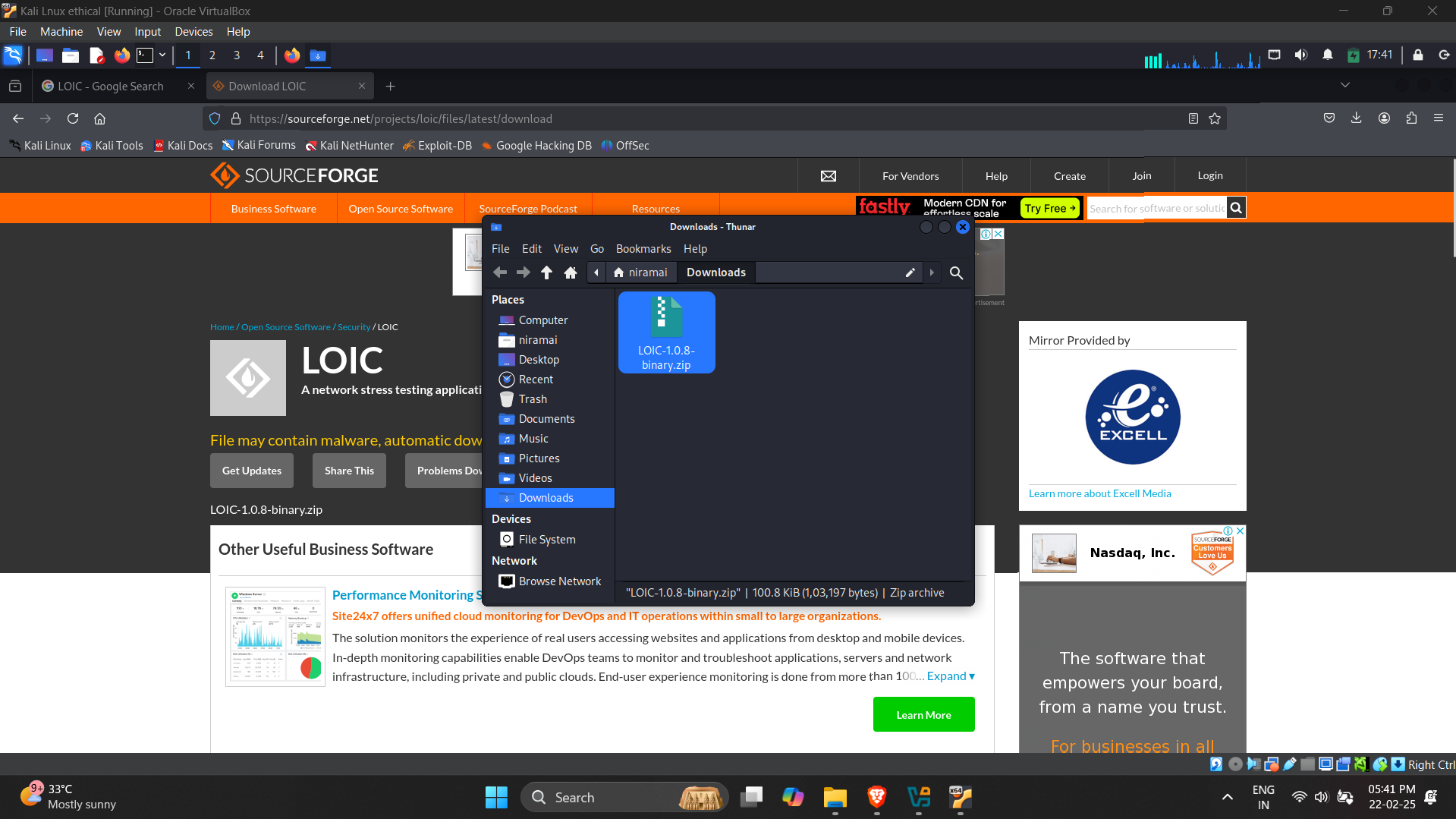
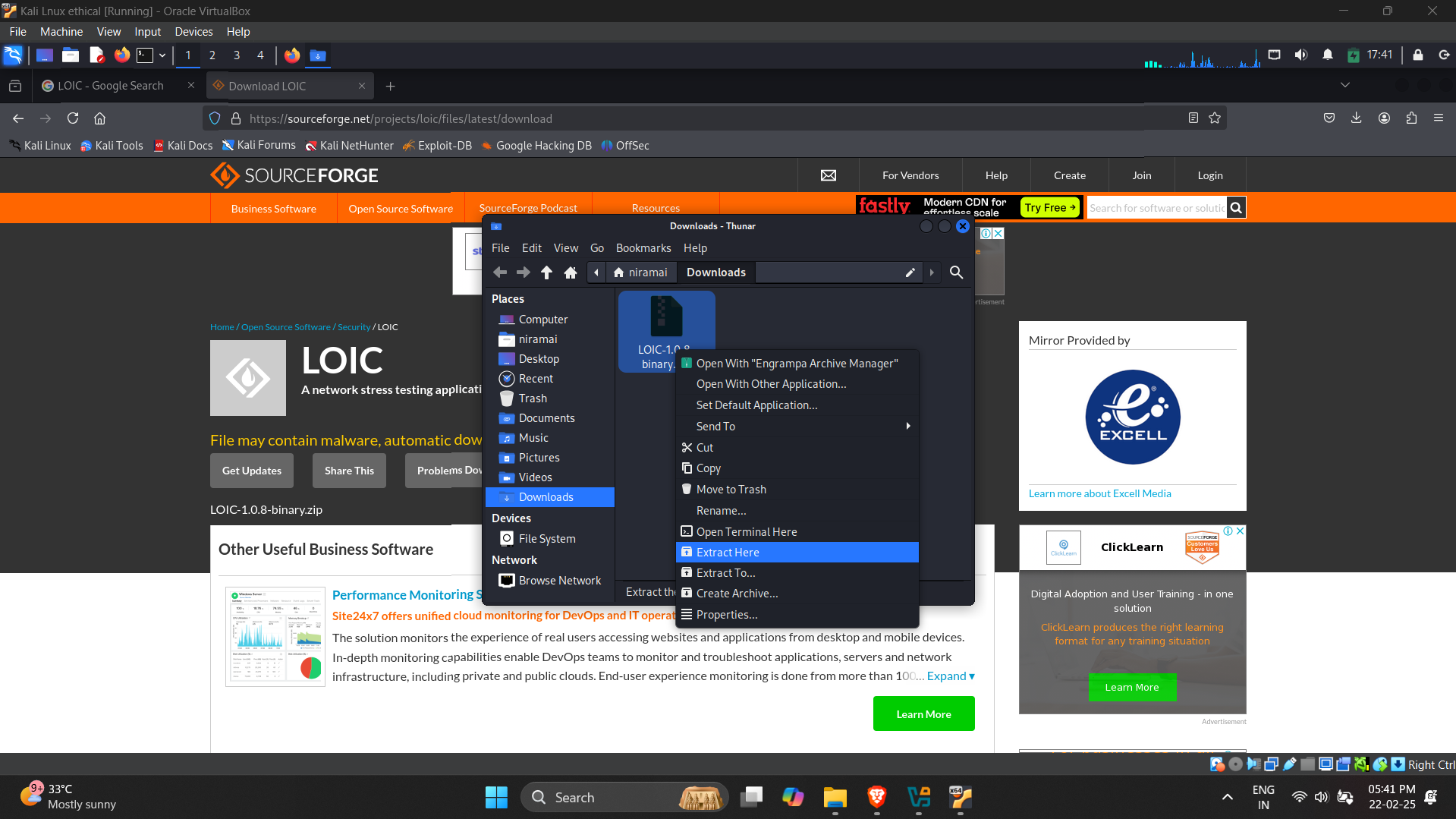
LOIC ->  
  
  
Installation -->

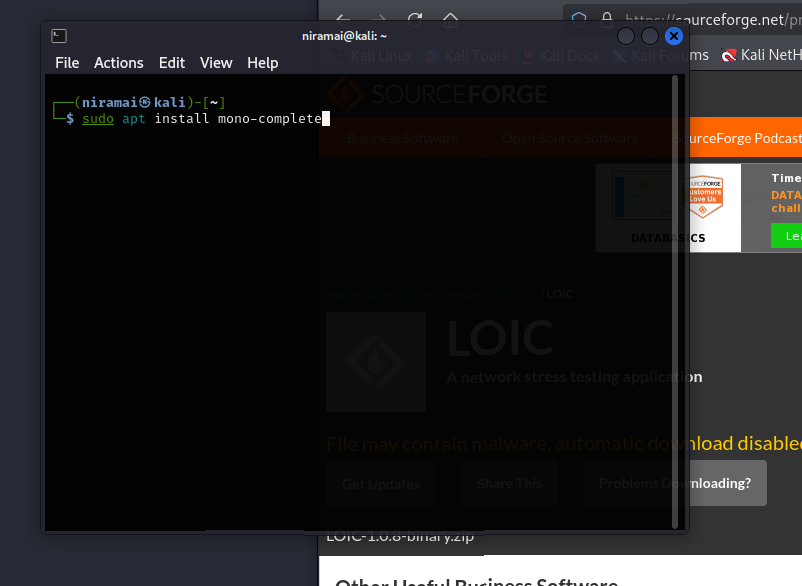




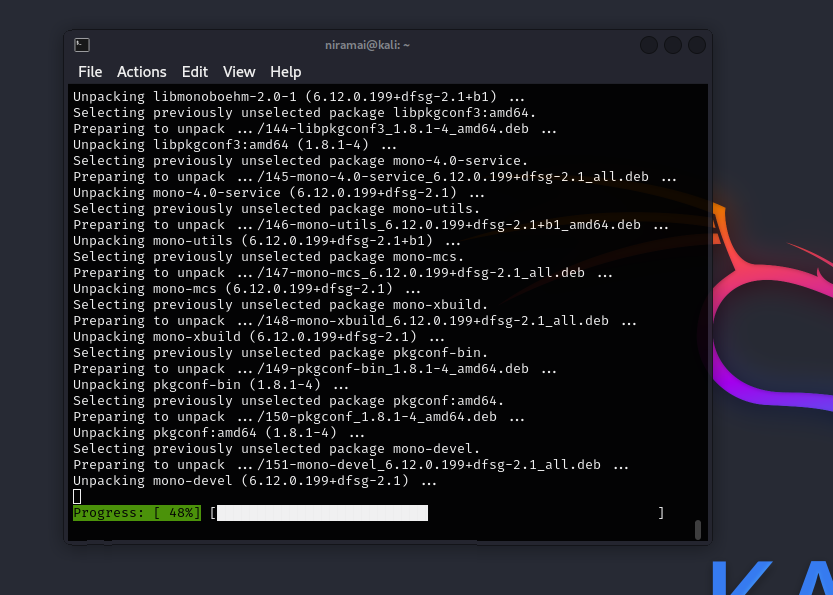


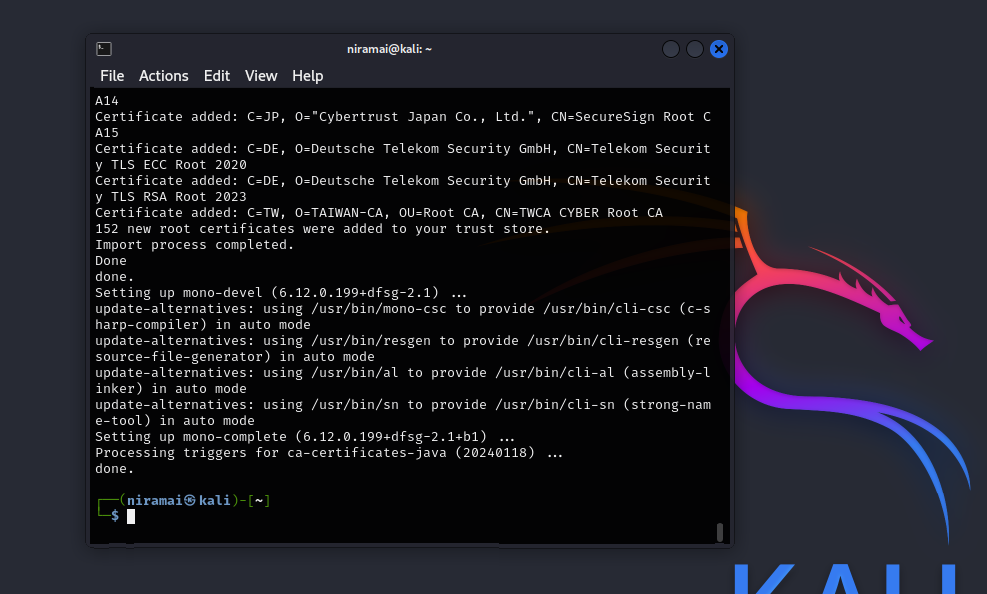




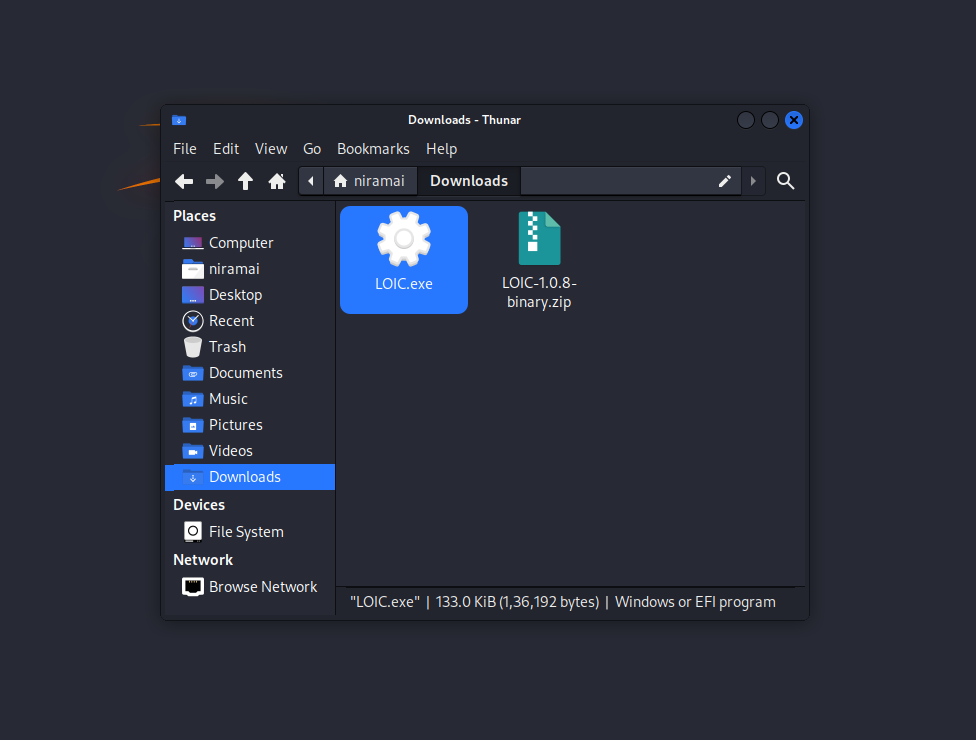


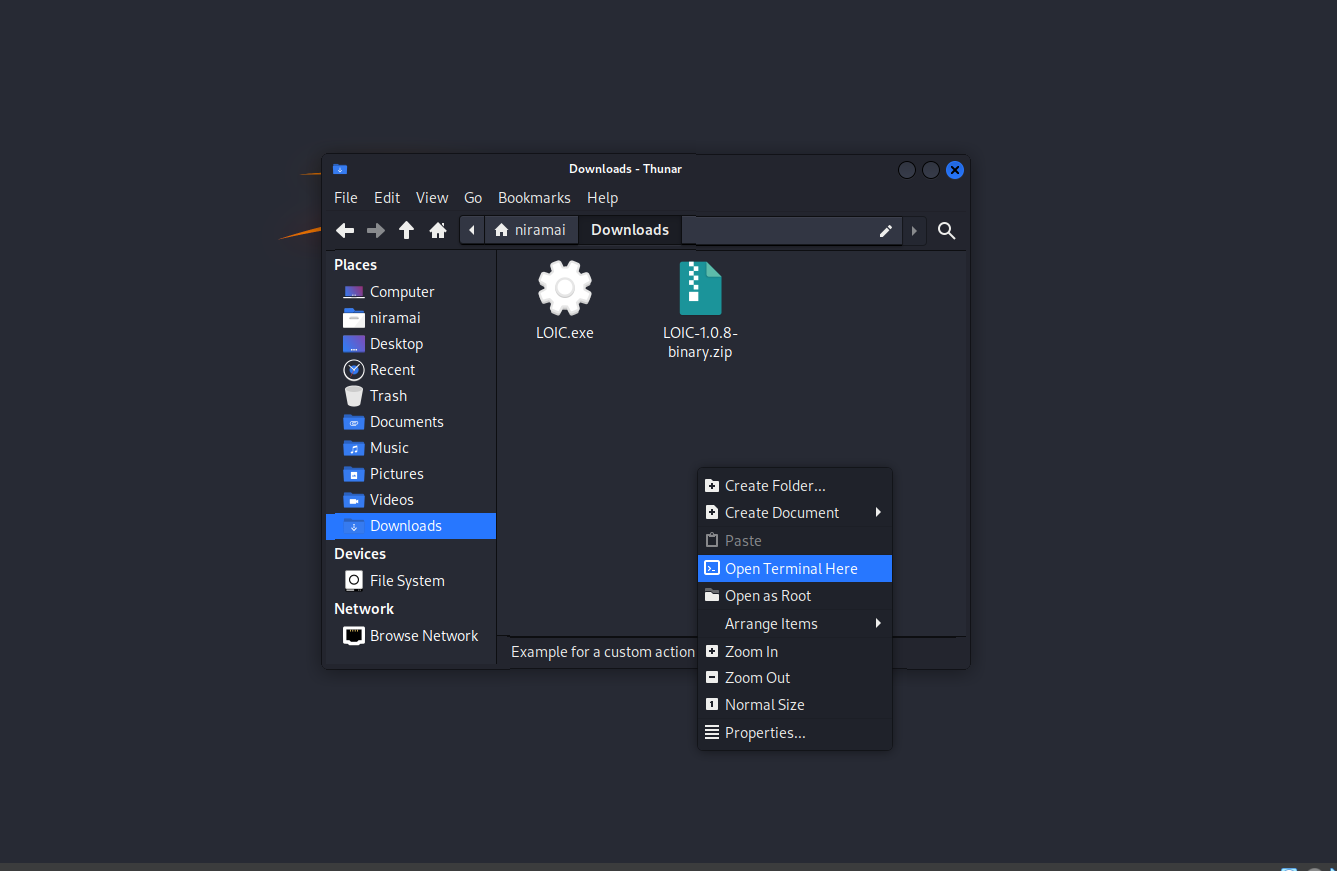
Sometimes if it doesnt run ask deepseek , its better than gpt in kali debugging .

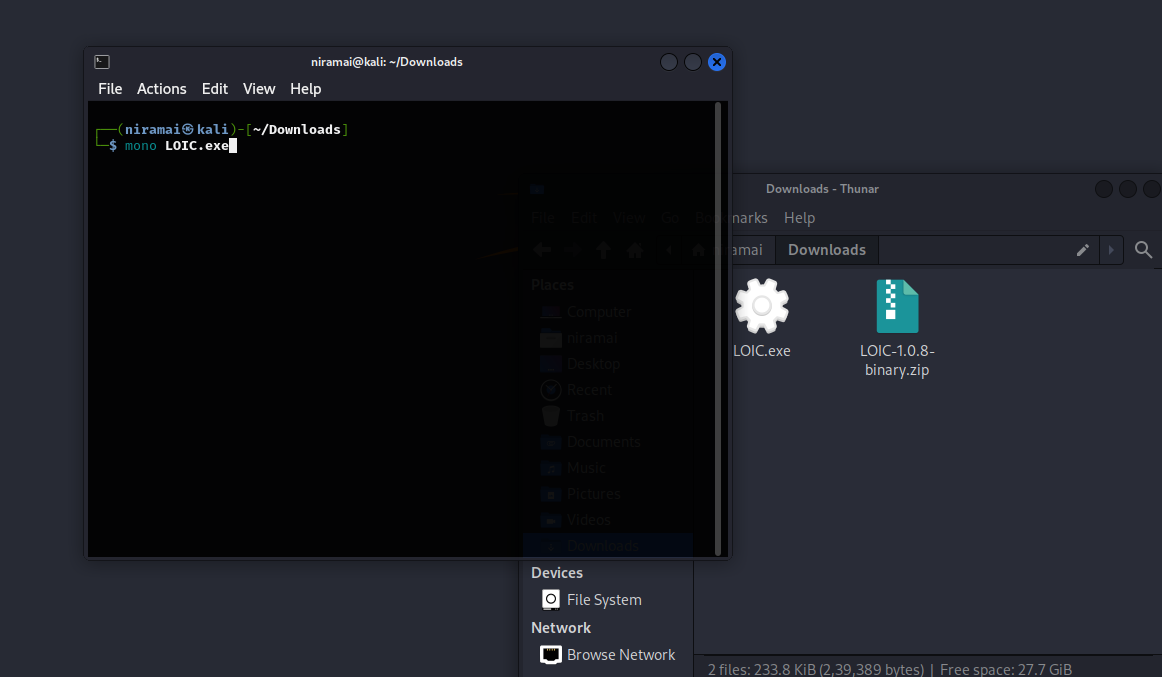


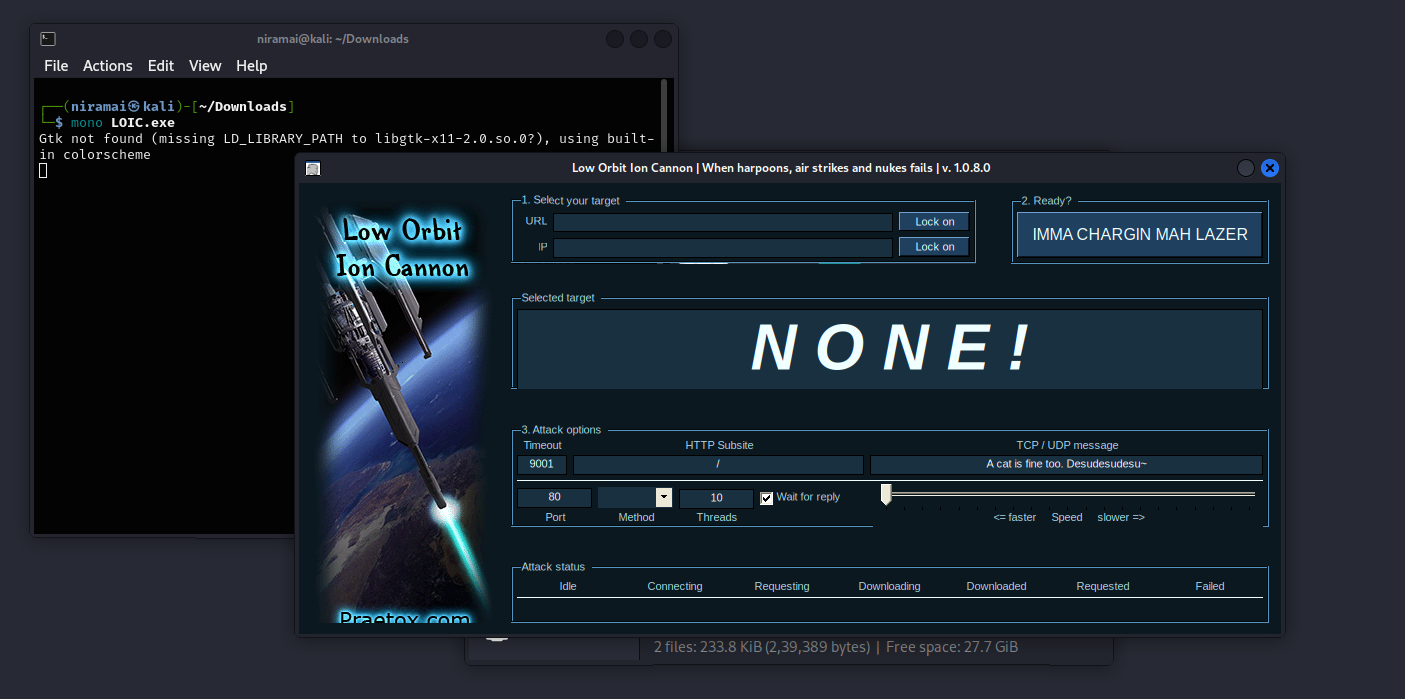


Locate the folder where LOIC is there









Usage :

Just type [www.google.com](https://www.google) and open sudo wireshark to check the messages

Or type in your IP

How Does a LOIC Work?

A LOIC works by flooding a target server with TCP, UDP, or HTTP packets with the goal of disrupting service. To make coordinated attacks easier, users can use IRC chat channels to run a “hivemind” version of the LOIC, allowing one primary user to control several networked secondary computers, creating a voluntary botnet. This is a popular approach because owners of the secondary devices can claim they were innocent victims of an involuntary botnet.

Notable LOIC Attack Instances and Events

The LOIC gained notoriety for its use by members of the hacktivist group Anonymous as well as users of the 4Chan forums. It was used during Project Chanology to attack websites for the Church of Scientology and the Recording Industry Association of America. It was also used during Anonymous' Operation Payback to attack the websites of companies and organizations that opposed WikiLeaks. To mitigate or prevent LOIC attacks, security experts have suggested that well-written firewall rules can filter out most traffic from DDoS attacks by LOIC, thus preventing the attacks from being fully effective

What is a Low Orbit Ion Cannon (LOIC) Tool?

The Low Orbit Ion Cannon (LOIC) is an open-source network stress testing application, often used by malicious actors and activists for denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks. It was developed in 2010 by Praetox Technologies and then released into the public domain. The tool works by flooding the target with TCP/UDP packets that target the network layer or HTTP GET requests that target the application layer. A more advanced version of the tool also exists called the High Orbit Ion Cannon (HOIC). There are two versions of the tool: the first is the binary version

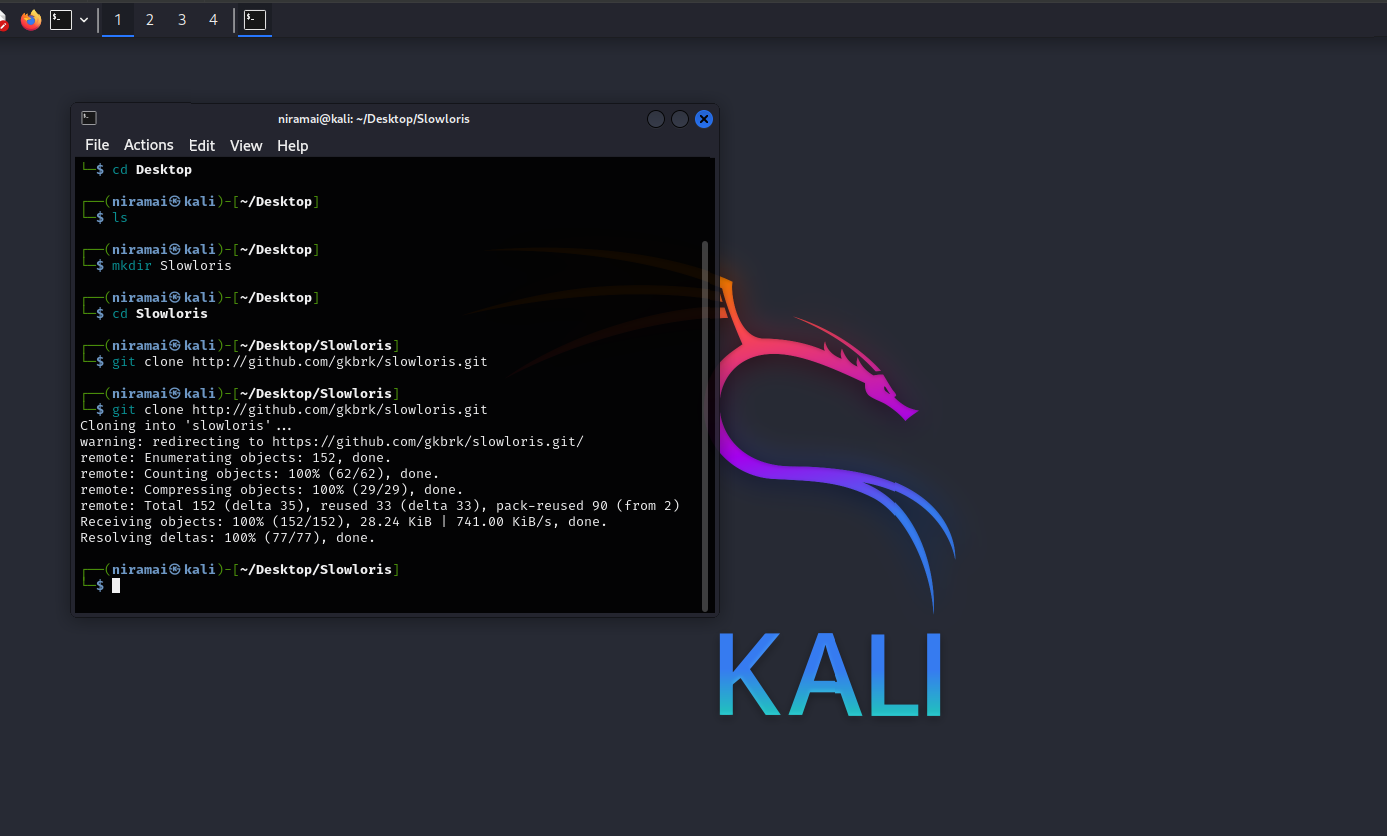
**Description :**

LOIC was used by Anonymous (a group that spawned from the /b/ board of 4chan) during Project Chanology to attack websites from the Church of Scientology, once more to (successfully) attack the Recording Industry Association of America's website in October 2010, and it was again used by Anonymous during their Operation Payback in December 2010 to attack the websites of companies and organizations that opposed WikiLeaks.

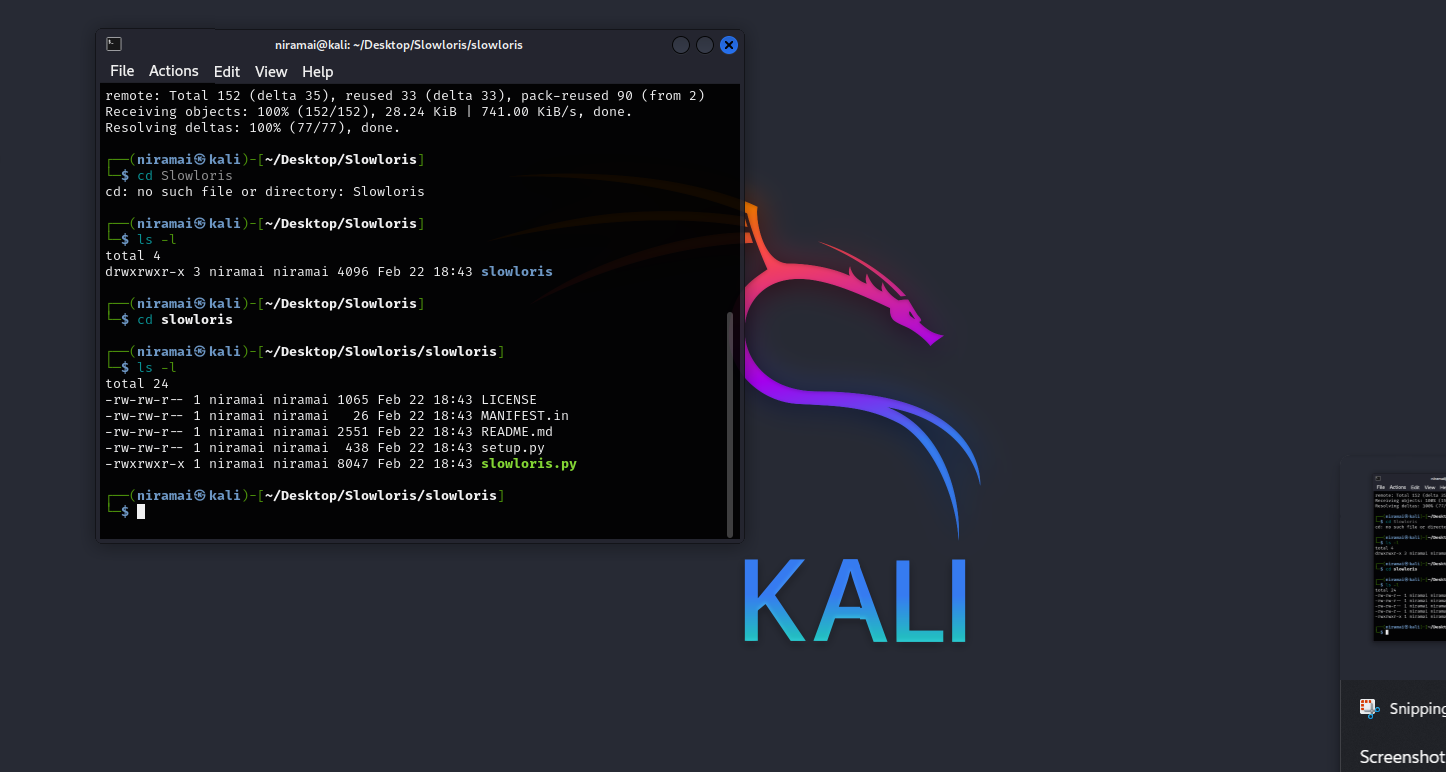
In retaliation for the shutdown of the file sharing service Megaupload and the arrest of four workers, members of Anonymous launched a DDoS attack upon the websites of Universal Music Group (the company responsible for the lawsuit against Megaupload), the United States Department of Justice, the United States Copyright Office, the Federal Bureau of Investigation, the MPAA, Warner Music Group and the RIAA, as well as the HADOPI, all on the afternoon of January 19, 2012, through LOIC. In general, the attack hoped to retaliate against those who Anonymous members believed harmed their digital freedoms.

**Slowloris:**

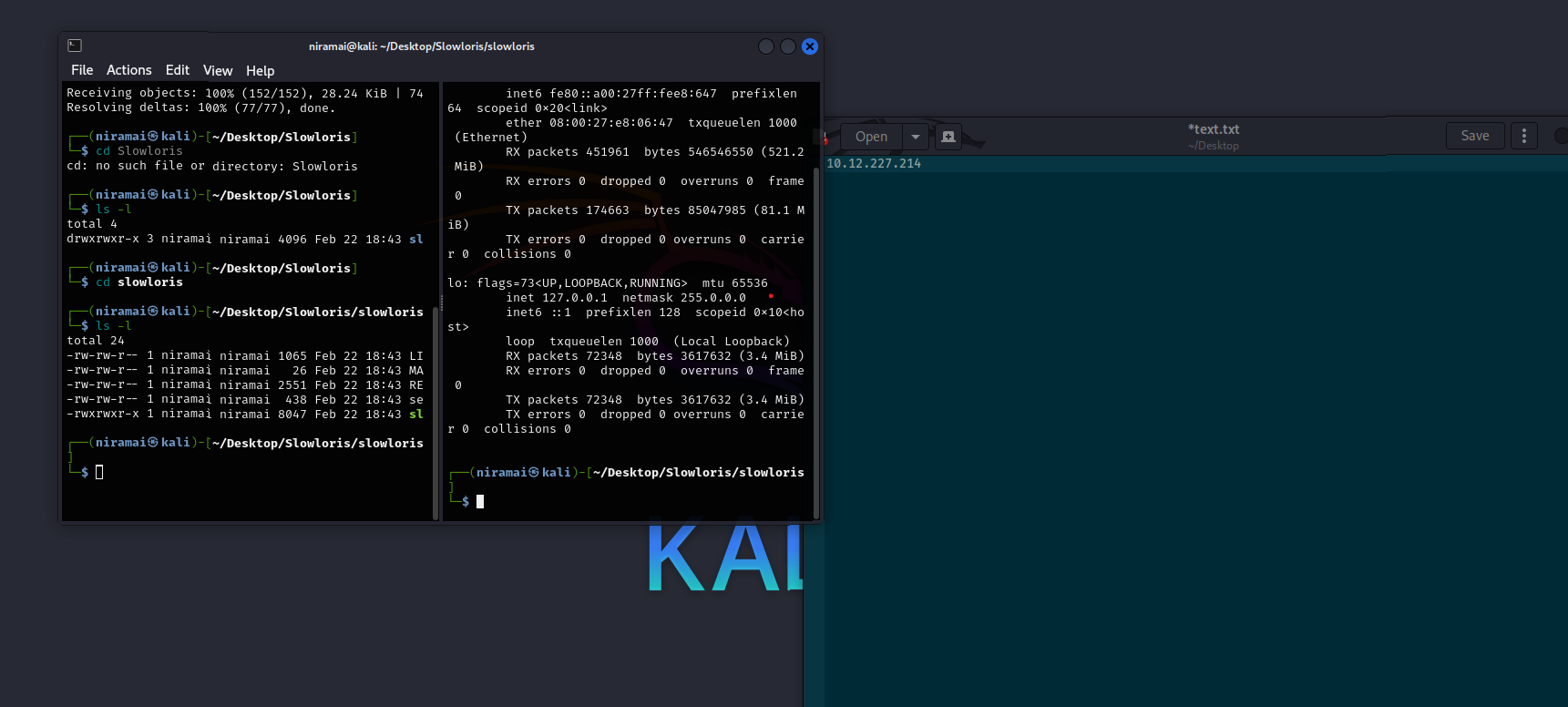
**Installation :**

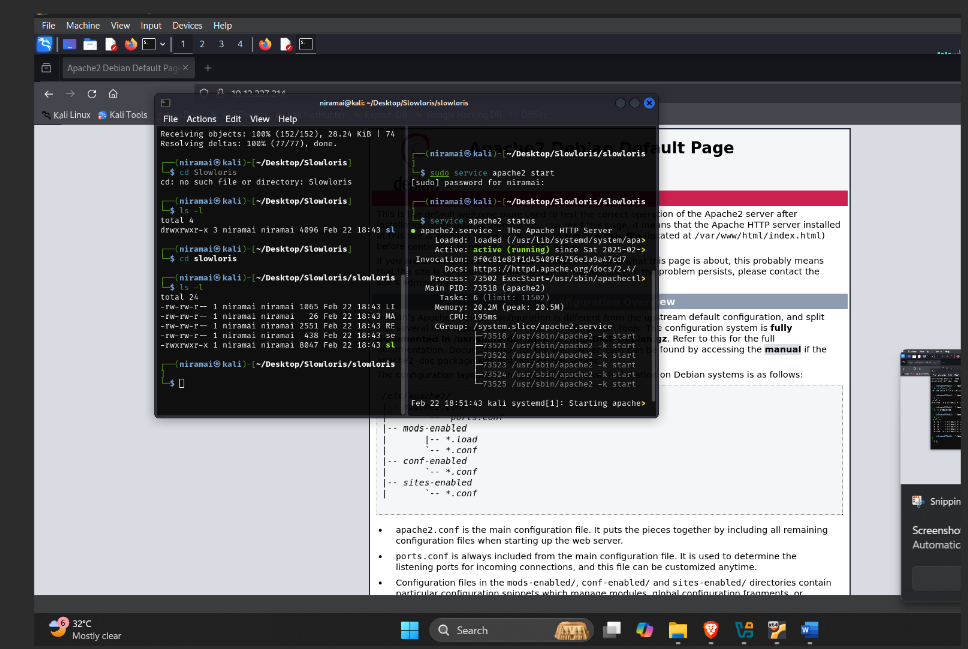


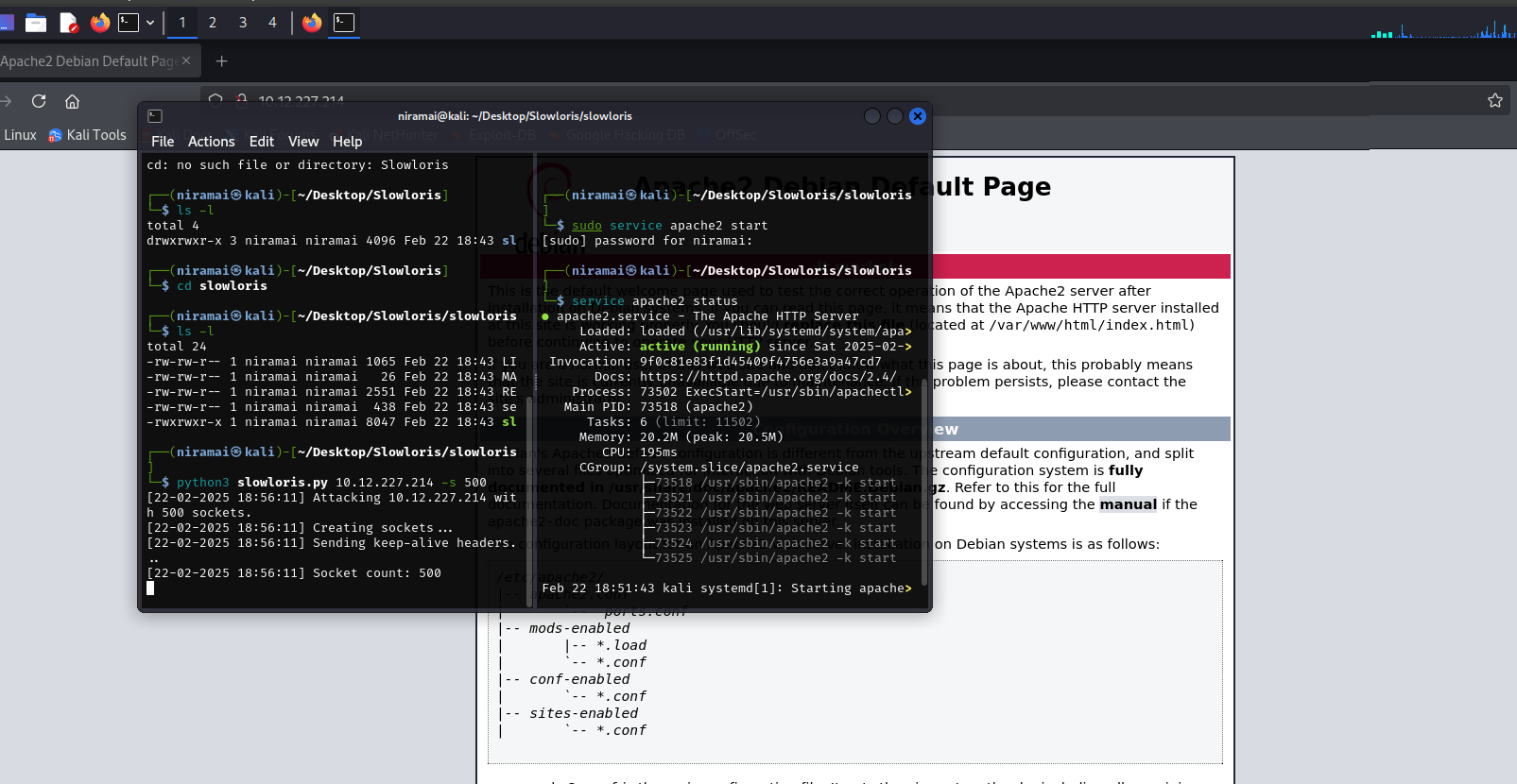
**git clone http://github.com/gkbrk/slowloris.git**



Actions --> split terminal vertically







Then u try reloading the tab , it will take a bit more time .   
  
A **Slowloris** DDoS attack is considered a distributed denial of service, and it can remain undetected by traditional intrusion detection systems by sending legitimate HTTP request packets at low request-per-second rates, rather than large volumes or high rates of HTTP requests per second. Additionally, since the log file cannot be written until a request is completed, **Slowloris** can immobilize a server for periods of time without a single entry appearing in the log file to raise a red flag for anyone monitoring it.

**Slowloris** attacks can be mitigated by: Limiting the number of connections a single IP address may request to open. Increasing the minimum transfer speed allowed for any connection. Limiting the time a client is allowed to stay connected. Increasing the maximum number of clients the server will allow. Deploying robust cloud mitigation services, configuring robust load balancers, using web application firewalls (WAFs) or other virtual patching techniques, and rate-limiting the number of requests per source.

What Are the Signs of a Slowloris DDoS Attack?

Much as its name implies, a Slowloris DDoS attack is slow and methodical. The attack involves sending partial HTTP requests to the targeted web server, with none ever being completed. As a result, the targeted server opens more connections, assuming the requests will be completed.Eventually, the server’s maximum allotted connection sockets are consumed one-by-one until fully exhausted, thus blocking any legitimate connection attempts. High-volume Web sites may take longer for Slowloris to completely take over, but ultimately the DDoS attack will result in all legit requests being denied.

How is a Slowloris attack conducted?

A Slowloris attack is executed in four steps: The attacker issues command and control instructions to their botnet or reverse proxies, which sends multiple HTTP requests, randomized headers, and other bypassing techniques, while opening multiple connections to a targeted server from their global networks. The targeted server opens a TCP connection for each request, planning to close the thread as soon as the HTTP request is completed. The server will time out any connection that is excessively long, to free up the thread for subsequent requests.

Slowloris git hub --> <https://github.com/gkbrk/slowloris>

Configuration options

It is possible to modify the behaviour of slowloris with command-line arguments. In order to get an up-to-date help document, just run slowloris -h.

-p, --port

Port of webserver, usually 80

-s, --sockets

Number of sockets to use in the test

-v, --verbose

Increases logging (output on terminal)

-ua, --randuseragents

Randomizes user-agents with each request

-x, --useproxy

Use a SOCKS5 proxy for connecting

--https

Use HTTPS for the requests

--sleeptime

Time to sleep between each header sent