

Definição do Trabalho Final

Disciplina: Laboratório de Redes de Computadores;

Professora: Cristina Moreira Nunes;

Alunos: Ana Cristina Follmann Schmidt, Bruno Bavaresco Zaffari, Ícaro Cecconello
Espadim, Lucas Pereira Salbego;

Introdução:

O trabalho tem por tarefas executar dois ataques explorando vulnerabilidades de protocolos da pilha TCP/IP (Sync flood e DoS HTTP), com ferramentas escolhidas pelo grupo em um servidor http. E, como ferramenta de detecção, desenvolver um sniffer que identifique os ataques. Usaremos o sistema operacional LINUX-KALI.

Sync flood:

Um ataque SYN Flood é uma técnica usada para sobrecarregar um servidor, fazendo com que ele fique indisponível para os usuários. Vou explicar como isso acontece, usando uma analogia comum. Imagine que você está em uma lanchonete, e cada cliente que chega pede para ser atendido, mas assim que o atendente começa a preparar o pedido, o cliente desaparece sem completar o pedido. Isso continuaria até que o atendente estivesse tão ocupado atendendo essas solicitações falsas que não poderia atender nenhum cliente real.

Em termos técnicos, no ataque SYN Flood, o atacante envia uma grande quantidade de pacotes SYN (uma parte do processo de handshake TCP, que é como um pedido para iniciar uma conexão) para um servidor. Normalmente, quando um servidor recebe um pacote SYN, ele responde com um pacote SYN-ACK (acknowledgement) e aguarda um pacote ACK de volta para estabelecer a conexão. No entanto, no ataque, o atacante nunca envia o pacote ACK de volta. Em vez disso, continua enviando mais e mais pacotes SYN, muitas vezes de endereços IP forjados.

Isso sobrecarrega o servidor, que mantém abertas todas essas conexões "meio iniciadas". Cada conexão que o servidor tenta manter aberta consome recursos. Eventualmente, se houver pacotes SYN suficientes, o servidor não poderá mais responder a solicitações legítimas, pois todos os seus recursos estão amarrados tentando completar essas conexões falsas. Isso pode levar ao que é conhecido como negação de serviço, onde o servidor fica tão sobrecarregado que efetivamente para de funcionar para usuários legítimos.

A defesa contra ataques SYN Flood geralmente envolve a identificação e o descarte de solicitações falsas, mas isso pode ser desafiador, pois os pacotes SYN falsos são frequentemente indistinguíveis dos legítimos. Além disso, técnicas como aumentar a capacidade de armazenamento de conexões incompletas, reduzindo o tempo que o servidor aguarda por um ACK, e usando listas de bloqueio de IP também são estratégias comuns para mitigar esses ataques.

DoS HTTP:

Um ataque DoS HTTP (Denial of Service) é uma forma de ataque cibernético que visa tornar um site ou serviço online indisponível para seus usuários. Essencialmente, ele sobrecarrega o servidor com um volume excessivo de solicitações de acesso, impedindo o funcionamento normal do serviço.

Durante um ataque DoS HTTP, o atacante utiliza múltiplas fontes para enviar uma enxurrada de requisições HTTP para um servidor web. HTTP, que significa HyperText Transfer Protocol, é o método pelo qual os dados são transferidos na internet. Cada solicitação HTTP que um servidor recebe exige que ele use recursos para responder. Normalmente, isso não é um

problema, mas em um ataque DoS, o número de solicitações é tão grande que o servidor não consegue processar todas, resultando em lentidão ou falha total.

O impacto de um ataque DoS HTTP pode variar. Em alguns casos, o site pode apenas ficar mais lento, enquanto em outros, pode ficar completamente inacessível. Isso acontece porque o servidor está tão ocupado tentando responder às solicitações falsas que não consegue atender aos usuários legítimos.

Para se defender contra ataques DoS HTTP, os administradores de sites utilizam várias estratégias. Isso inclui identificar e filtrar o tráfego suspeito, distribuir a carga entre vários servidores para diluir o impacto das solicitações e utilizar sistemas de proteção contra ataques cibernéticos. Estes sistemas são projetados para reconhecer e mitigar ataques, garantindo que o serviço continue disponível para os usuários legítimos.

Diferenças:

Um ataque SYN Flood foca na camada de transporte da rede, inundando um servidor com solicitações de conexão iniciais (SYN) para esgotar seus recursos, enquanto um ataque DoS HTTP opera na camada de aplicação, sobrecarregando o servidor com um grande volume de solicitações HTTP, que são pedidos de acesso a conteúdo da web. Enquanto o SYN Flood explora o processo de estabelecimento de conexão TCP, o DoS HTTP visa a incapacidade do servidor de processar inúmeras solicitações de conteúdo simultâneas.

Ferramentas usadas para o ataque:

Utilizamos para realizar o trabalho as ferramentas hping3, para realização do SYNC FLOOD e LOIC (Low Orbit Ion Cannon), para a realização do DoS HTTP.

hping3:

O hping3 é uma ferramenta poderosa usada em redes de computadores para testes de segurança e análise de tráfego. No contexto de um ataque SYN flood, o hping3 pode ser configurado para enviar uma grande quantidade de pacotes SYN (uma parte do processo de handshake TCP) para uma máquina alvo. Aqui está como funciona:

Quando você inicia um ataque SYN flood usando hping3, a ferramenta gera uma inundação de solicitações de conexão TCP (os pacotes SYN) para o servidor alvo. Esses pacotes são enviados rapidamente e em grande quantidade. Cada pacote carrega um endereço de IP de origem falsificado, tornando difícil para o servidor identificar e bloquear a fonte do ataque. Normalmente, quando um servidor recebe uma solicitação de conexão TCP, ele responde com um pacote SYN-ACK e aguarda a confirmação final do cliente para completar o processo de handshake e estabelecer a conexão. Mas, no ataque SYN flood, como os endereços de origem são falsificados e nunca respondem ao SYN-ACK, o servidor acaba esperando indefinidamente por muitas respostas que nunca chegarão. Isso leva à saturação dos recursos do servidor, pois ele mantém essas "semi-conexões" abertas, esperando por respostas que nunca virão, eventualmente sobrecarregando o servidor e podendo até levá-lo a um estado de inoperância.

Essa técnica é uma forma de ataque de negação de serviço (DoS), onde o objetivo não é ganhar acesso ao sistema, mas sim tornar o serviço inacessível para usuários legítimos.

LOIC:

O LOIC, que significa Low Orbit Ion Cannon, é outra ferramenta usada para realizar ataques de negação de serviço (DoS) ou distribuídos de negação de serviço (DDoS). Originalmente

desenvolvida para testar a resistência das redes, ela se tornou popular entre os usuários com menos conhecimento técnico devido à sua interface simples e fácil de usar.

O funcionamento do LOIC é bastante direto. Quando você executa um ataque usando o LOIC, a ferramenta envia uma grande quantidade de solicitações de rede para um servidor alvo. Isso é feito para sobrecarregar a capacidade do servidor de lidar com múltiplas solicitações simultâneas, o que pode levar a interrupções no serviço. O LOIC permite que o usuário defina o endereço IP do servidor alvo, a porta sobre a qual o ataque será realizado, e o tipo de protocolo a ser usado (TCP, UDP ou uma requisição HTTP).

Um dos aspectos mais notáveis do LOIC é a facilidade com que ele pode ser utilizado para lançar um ataque DDoS. Em um ataque DDoS, várias máquinas são usadas para atacar um único alvo, aumentando significativamente a magnitude do ataque comparado a um ataque DoS simples, que origina de uma única fonte. O LOIC foi popularmente usado em ataques coordenados, onde vários usuários voluntariamente conectavam suas máquinas a uma rede de comando e controle, permitindo que seus computadores fossem usados para amplificar o ataque.

Ferramenta HTTP Server:

Utilizamos como alvo dos ataques o Apache2.

O Apache 2 é um servidor web amplamente utilizado, conhecido por sua flexibilidade e poder. Funciona hospedando e servindo páginas da web, que podem ser simples páginas HTML ou podem incluir scripts complexos em linguagens como PHP ou Python. Quando um usuário acessa um site, o navegador envia uma solicitação HTTP ao servidor Apache, que então processa a solicitação e retorna a página da web solicitada.

Em um contexto de teste de segurança em uma rede local, podemos utilizar o Apache 2 como um alvo para simular ataques. Isso é útil para entender como o servidor se comporta sob carga ou tentativas de intrusão, permitindo aos administradores de rede avaliarem e melhorar as medidas de segurança. Por exemplo, podemos usar ferramentas como o hping3 ou o LOIC para realizar ataques de DoS ou DDoS ao Apache 2, monitorando como o servidor responde ao excesso de solicitações. Essa abordagem ajuda a identificar vulnerabilidades, como a capacidade do servidor de lidar com grandes volumes de tráfego, e a eficácia de eventuais medidas de segurança, como firewalls ou sistemas de detecção e prevenção de intrusões.

Além disso, usar o Apache 2 em um ambiente controlado, como uma rede local, é uma prática segura para tais testes, pois evita os riscos legais e éticos associados ao lançamento de ataques contra servidores reais na internet. É importante lembrar que qualquer teste de segurança deve ser realizado em um ambiente controlado e com a devida autorização, para não infringir leis ou causar danos não intencionais a sistemas de terceiros. Essa prática não só é crucial para a aprendizagem e compreensão dos estudantes sobre a segurança cibernética, mas também para a aplicação responsável desses conhecimentos.

USANDO E INSTALANDO OS SOFTWARES:

USANDO O APACHE2:

Para instalar, no sistema Linux-Kali:

Passo 1: Primeiro, atualize os pacotes do seu sistema para garantir que você tenha as versões mais recentes. Abra um terminal e execute: `sudo apt update`;

Passo 2: Após a atualização, instale o Apache 2 usando o seguinte comando: `sudo apt install apache2`;

Passo 3: Uma vez instalado, você pode verificar o status do Apache para garantir que ele está rodando corretamente. Use o comando: `sudo systemctl status apache2`;

Passo 4: Se você estiver usando um firewall, precisará permitir o tráfego HTTP. No Kali Linux, o UFW (Uncomplicated Firewall) é comumente usado. Para permitir o tráfego HTTP, execute: `sudo ufw allow in "Apache"`;

Após instalado, tem esses comandos a serem executados no terminal:

1. Use para começar o apache2: `sudo systemctl start apache2`

```
(brunobavarescozaffari@brunobavarescozaffar)~]
$ sudo systemctl start apache2

[sudo] password for brunobavarescozaffari:
```

2. Use para verificar o status do apache2: `sudo systemctl status apache2`

```
(brunobavarescozaffari@brunobavarescozaffar)~]
$ sudo systemctl status apache2

● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; disabled; preset: >
   Active: active (running) since Sat 2023-11-25 17:01:16 -03; 54s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 3918 ExecStart=/usr/sbin/apachectl start (code=exited, status=0>
  Main PID: 3934 (apache2)
    Tasks: 6 (limit: 3461)
   Memory: 19.6M
      CPU: 57ms
   CGroup: /system.slice/apache2.service
           └─3934 /usr/sbin/apache2 -k start
             └─3937 /usr/sbin/apache2 -k start
               └─3938 /usr/sbin/apache2 -k start
                 └─3939 /usr/sbin/apache2 -k start
                   └─3940 /usr/sbin/apache2 -k start
                     └─3941 /usr/sbin/apache2 -k start

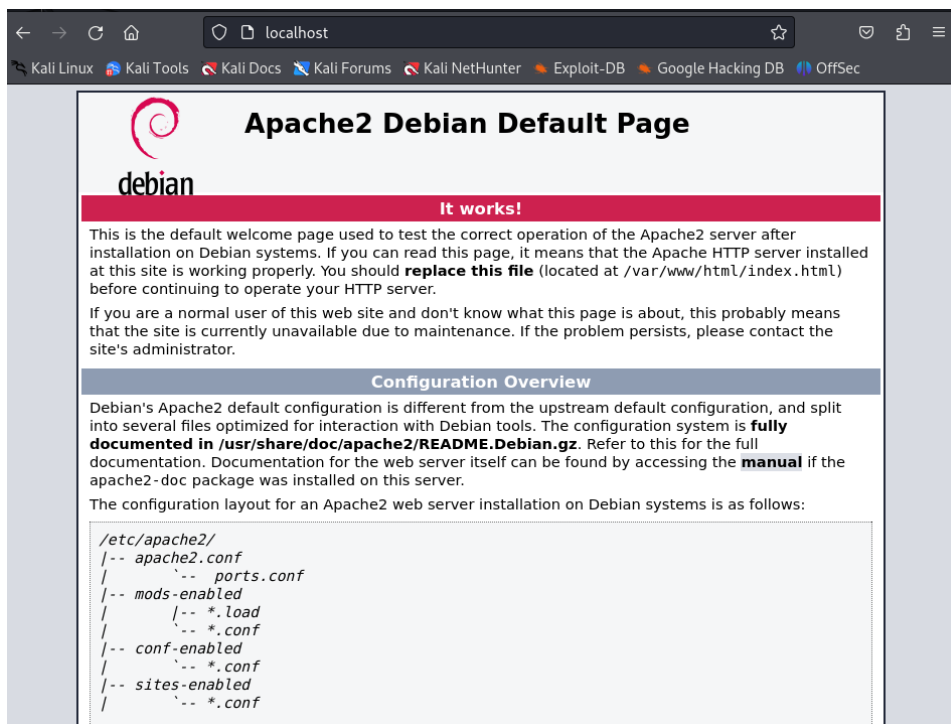
Nov 25 17:01:15 brunobavarescozaffar systemd[1]: Starting apache2.service - >
Nov 25 17:01:16 brunobavarescozaffar systemd[1]: Started apache2.service - T>
lines 1-19/19 (END) ... skipping ...
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; disabled; preset: disable>
```

3. Use para parar o apache2: `sudo systemctl stop apache2`

```
(brunobavarescozaffari@brunobavarescozaffar)~]
$ sudo systemctl stop apache2

[sudo] password for brunobavarescozaffari:
```

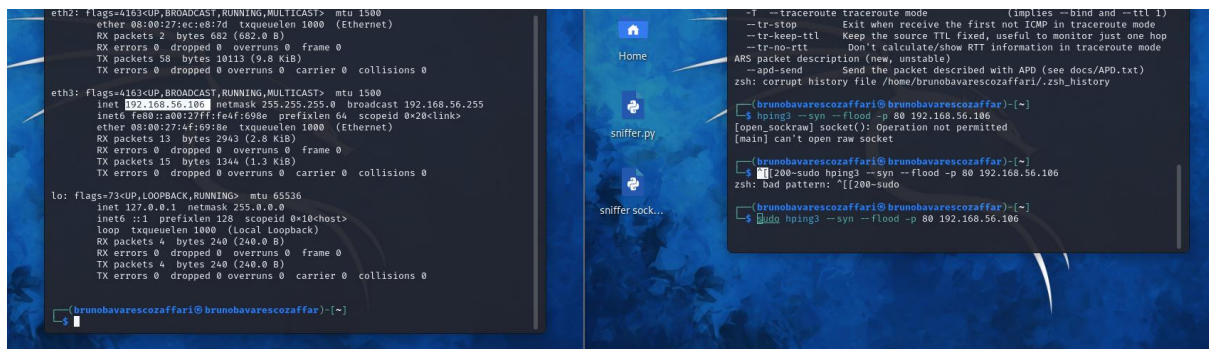
Para verificar a página, vá até o navegador e digite `http://localhost` e verá:



USANDO O HPING3:

Ele já vem instalado normalmente no Linux kali. Então, é somente usar com o terminal e no modo administrador.

Para usar é preciso saber o ip do servidor e executar o comando: `sudo hping3 --syn --flood -p 80 <IP QUE SERÁ ATACADO>`




USANDO O LOIC:

Ele não vem no sistema KALI-LINUX e precisa dar uns passos para a instalação do mesmo:

Passo 1: ir até o site : <https://sourceforge.net/projects/loic/files/latest/download> e baixar;



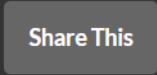
Home / Browse Open Source / Security / LOIC




LOIC

A network stress testing application

★★★★★ 94 Reviews Downloads: 5,003 This Week Last Update: 2020-08-17

Summary	Files	Reviews	Support	Donate 
---------	-------	---------	---------	--

Low Orbit Ion Cannon.

The project only keeps and maintains (bug fixing) the code written by the original author - Praetox, but is not associated or related to it.

DISCLAIMER: USE ON YOUR OWN RISK. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER OR CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,

Passo 2: Extraia .zip, do download, os arquivos em uma pasta.

Passo 3: Instale mono-complete: `sudo apt install mono-complete`;

```
File Actions Edit View Help
(kali@kali)-[~]
$ sudo apt install mono-complete
```

Passo 4: Para iniciar, vá até o diretório em que se encontra o executável, via terminal, nesse diretório, execute o comando: `mono LOIC.exe`;

```
(brunobavarescozaffari@brunobavarescozaffari)-[~/Downloads]
$ mono LOIC.exe

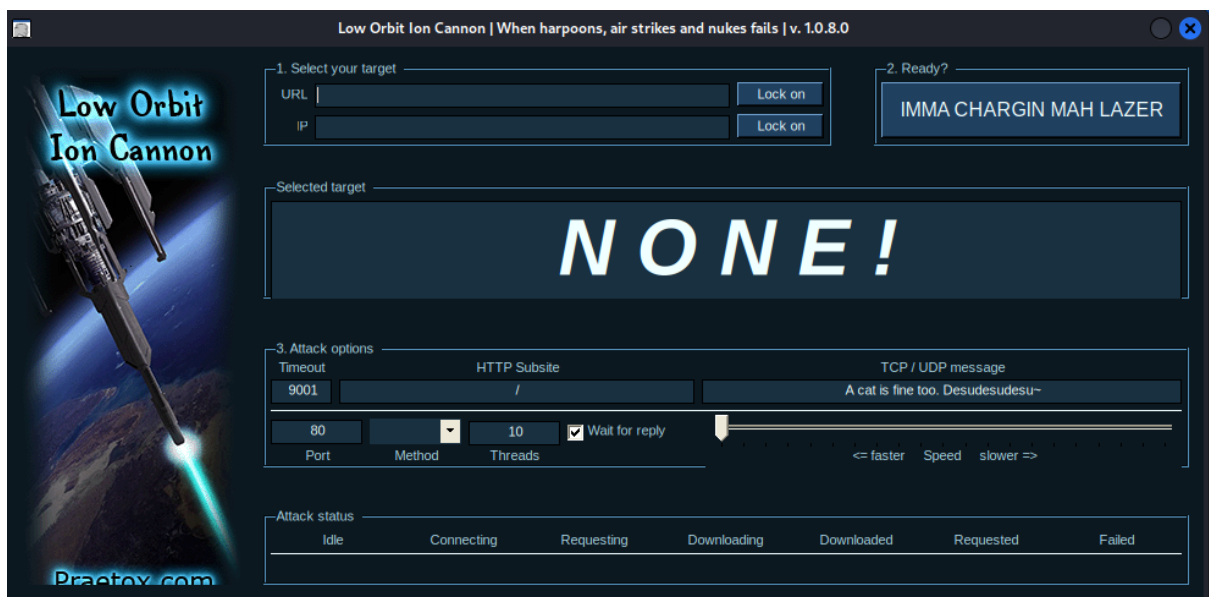
Native Crash Reporting

Got a SIGSEGV while executing native code. This usually indicates
a fatal error in the mono runtime or one of the native libraries
used by your application.

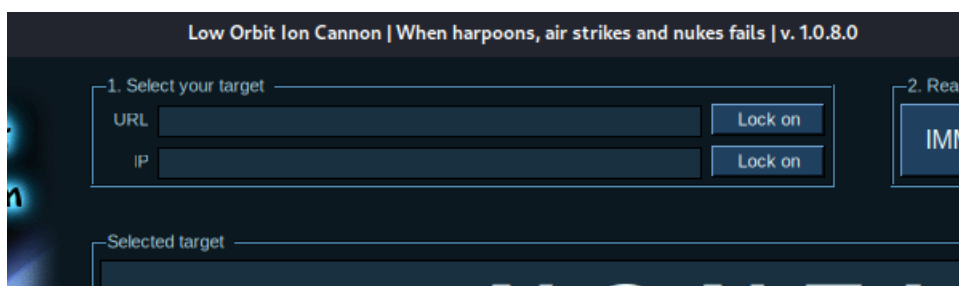
Native stacktrace:

0x55a21eaa2a8c - mono : (null)
0x55a21eaa2e30 - mono : (null)
0x55a21eaa5188 - mono : (null)
```

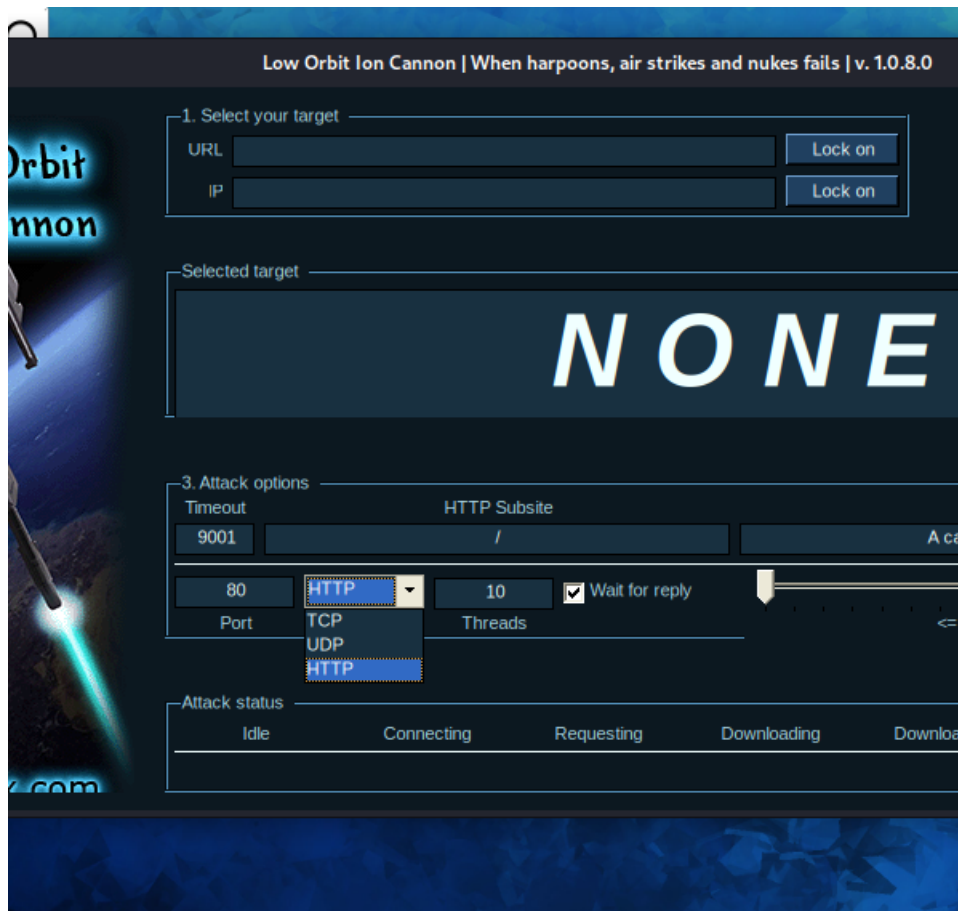
Passo 5: Uso



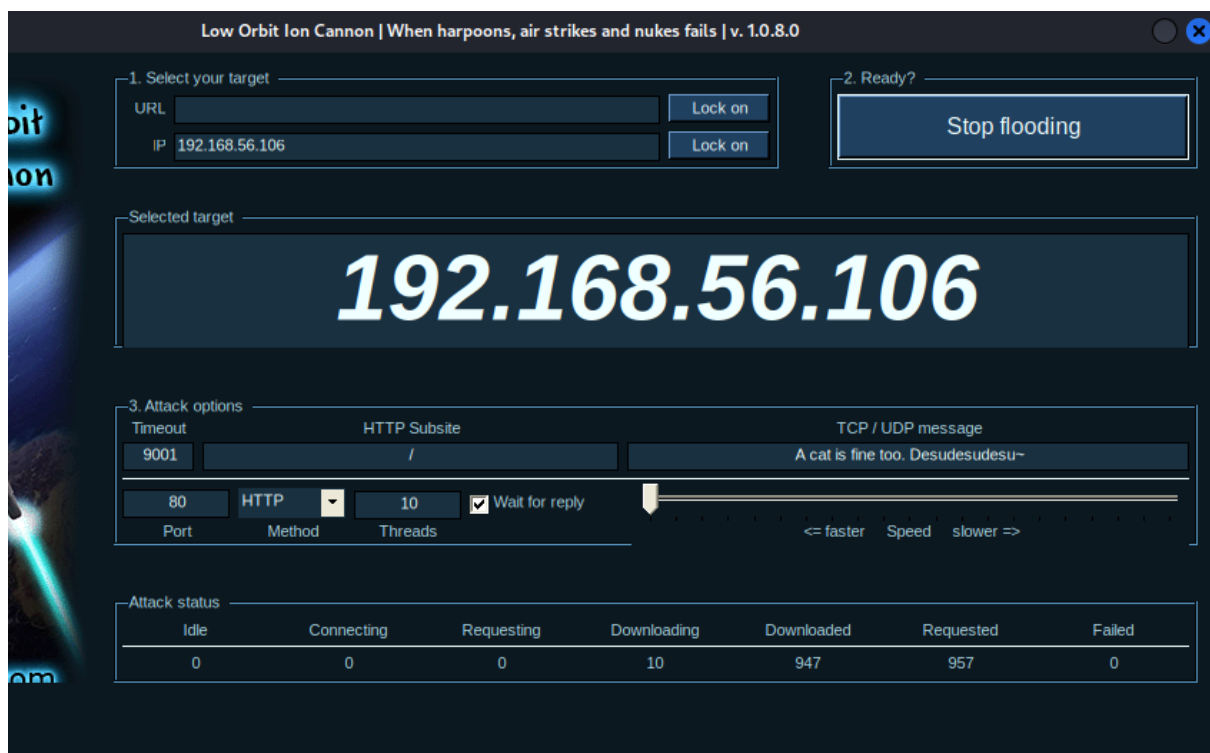
Ponha o URL ou o IP da máquina a ser atacada:



Aqui podemos selecionar qual protocolo queremos utilizar para o ataque:



Ao funcionar:



Ao realizar o ataque hping3, como pode ser identificado:

Ao realizar o ataque LOIC, como pode ser identificado: