Web Server 2 Ryan Fuller

1. What were your three benchmark workloads? Include the Python source code of your locust jobs.
   1. benchmark1.py (Users=100, Spawn=100):

from locust import HttpUser, task

class WebsiteUser(HttpUser):

@task(4)

def f100(self):

self.client.get("/file100.html")

@task(6)

def f1000(self):

self.client.get("/file1000.html")

This benchmark is to test how the server handles when the @task call is mentioned more than twice (i.e. task(4)). This is a relatively medium stress test on the server.

* 1. benchmark2.py (Users=100, Spawn=1000):

from locust import HttpUser, task

class WebsiteUser(HttpUser):

@task(6)

def f10000(self):

self.client.get("/file10000.html")

@task

def f100(self):

self.client.get("/file100.html")

The second benchmark is a low-stress test, seeing how the second self.client.get would behave when given a smaller file and called less.

* 1. benchmark3.py (Users=10000, Spawn=100000):

from locust import HttpUser, task

class WebsiteUser(HttpUser):

@task(50)

def f100000(self):

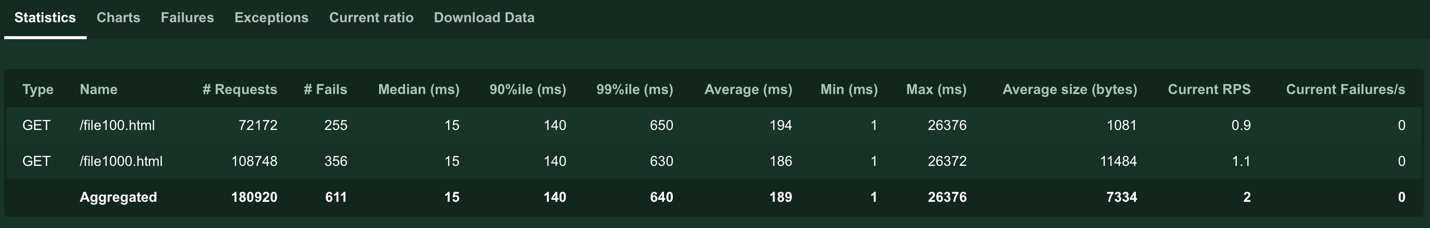
self.client.get("/file100000.html")

@task(100)

def f1000000(self):

self.client.get("/file1000000.html")

1. What was the performance of your basic implementation?
   1. benchmark1.py:

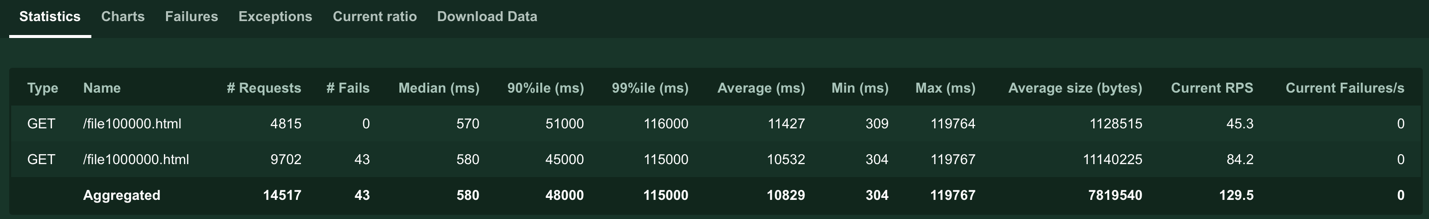


* 1. benchmark2.py:

A screenshot of a computer

Description automatically generated with medium confidence

* 1. benchmark3.py:



1. What was the performance impact of streaming?
   1. With streaming, there were more failures across all benchmark python file tests, but the RPM was generally higher, spitting out more requests faster. This means that streaming is more dangerous and will lead to a more unstable server when under load, but performs at a faster rate.
2. What was the performance impact of caching?
3. What conclusions do you draw from your experiments?
   1. It is safe to conclude that streaming files rather than reading the entire file into RAM does process requests faster but generally fails more often, making it less safe than its counterpart.