Import the Modules

```
import simpy as sy # Simpy Module to run simulations

import random # Running random probabilities
```

Creating the Parameters to Use for Simulation

```
random.seed(1920)
air_serv = 30 # Boarding Pass Checkers
air_scan = 20 # Boarding Pass Scanners
runTime = 1000 # times per simulation
total_time = 0
total_passengers = 1
sum_pass = 500 # Done the simulation with a selected number
```

Creating the Definitions to Run the Simulations

```
class Travel_Screen(object):
    """ID/boarding pass check queue and security scanning
    """
    def __init__(self, env, air_serv, air_scan):
        self.env = env
        self.server = sy.Resource(env, air_serv)
        self.scanner = sy.Resource(env, air_scan) # Running the
resources to serve the customer


    def IDcheck(self, passenger):
        """The ID check process. It takes a "person", checks their
ID,then
        passes him/her to the next step."""
        serv_time = random.expovariate(.75) # Mean Rate 2 (Check Rate)
        yield self.env.timeout(serv_time)

    def scan(self, passenger):
        """Security Scan. Passenergs are sent to shortest queue then
scanned"""
        scan_time = random.uniform(0.5, 1) # Going through the Personal
Scanner for Min Scan and Max Scan
        yield self.env.timeout(scan_time)
```

Creating the Passenger Simulation

```
def Passenger(env, number, t):
    """Passnegers arrive, to the first available server for the ID
check then
    sent to the first available scanner.
    """
    global total_time #global average wait time
    global total_passengers
    Arrivaltime = env.now
    with t.server.request() as request:
        yield request
        yield env.process(t.IDcheck(number))


    with t.scanner.request() as request:
        yield request
        yield env.process(t.scan(number))

    pass_time = env.now - Arrivaltime
    total_time = total_time + pass_time
    total_passengers = total_passengers+1
```

Setting Up

```
def replicate(env, num, tsa):
    arrival_int = random.expovariate(5)
    yield env.timeout(arrival_int)
    env.process(Passenger(env, num, tsa))
```

Running the Simulation

```
print('Getting Screened')
random.seed(245)
obs = sy.Environment()
ap = Travel_Screen(obs, air_serv, air_scan)
# Start processes and run
for i in range(0,sum_pass):
    obs.process(replicate(obs, i, ap))

obs.run()

avg_time = total_time / total_passengers

print("Average time to be screened:")
print(avg_time)
```

```
Getting Screened
Average time to be screened:
12.539123495250236
```

After calculating the average time for screening, it has been determined that by changing the parameters between 450-500 passengers, the average time gets much closer in the 10-15 minute range by having 30 checkers and 20 scanners. However, by testing 1000 passengers, the time has doubled to 20-30 minutes, which is expected. Although, by changing the parameters for the checkers and scanners, the more scanners there are, the slower the checkout time will be, even with less passengers.