```
Name: Laxmi Shejwal

Roll No: 17
```

A Call Center : ¶

call center, where employees attempt to cold call potential customers and get them to purchase our product.

```
A call center where each employee gets assigned with 50 calls each day on an
average.
```

```
The probability of conversion over each call is equal to 4%.
```

```
The average revenue generation for the company based on each such conversion is
that of USD 20.
```

```
If you analyze 100 such employees, who get paid USD 200 each day, then
n = 50

p = 4%
```

In [2]:

```
import numpy as np

# Call Center Simulation
# Number of employees to simulate
employees = 100

# Cost per employee
wage = 200

# Number of independent calls per employee
n = 50

# Probability of success for each call
p = 0.04

# Revenue per call
revenue = 100

# Binomial random variables of call center employees
conversions = np.random.binomial(n, p, size=employees)

# Print some key metrics of our call center
print('Average Conversions per Employee: ' + str(round(np.mean(conversions), 2)))

print('Standard Deviation of Conversions per Employee: ' + str(round(np.std(conve

print('Total Conversions: ' + str(np.sum(conversions)))

print('Total Revenues: ' + str(np.sum(conversions)*revenue))

print('Total Expense: ' + str(employees*wage))

print('Total Profits: ' + str(np.sum(conversions)*revenue - employees*wage))
```

```
Average Conversions per Employee: 2.17
Standard Deviation of Conversions per Employee: 1.38
Total Conversions: 217
Total Revenues: 21700
Total Expense: 20000
Total Profits: 1700
```

Recalling that each employee’s results follows a binomial distribution

In [19]:

```
import matplotlib.pyplot as plt
import seaborn as sns
```

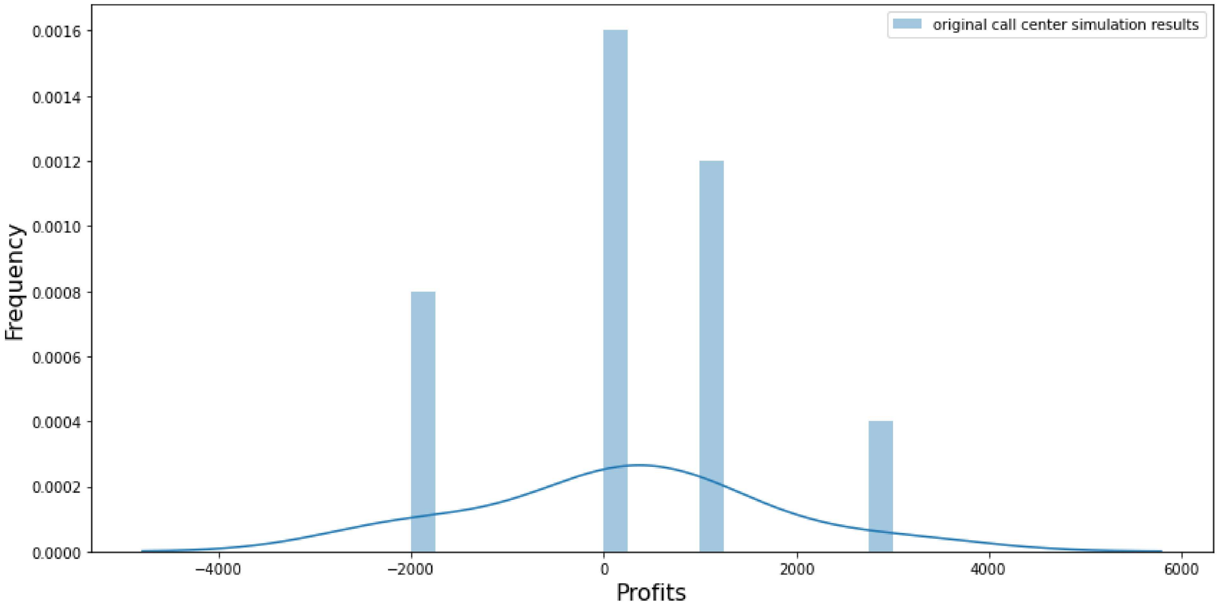
In [20]:

```
employees = 10
wage = 200
n = 50
p = 0.04
revenue = 1000
sims = 500


conversions = np.random.binomial(n, p, size=employees)


sim_profits = np.array(conversions)*revenue - employees*wage

sim_conversions_up = [np.sum(np.random.binomial(n, p, size=employees)) for i in r
sim_profits_up = np.array(sim_conversions_up)*revenue - employees*wage

# Plot and save the results as a histogra
fig, ax = plt.subplots(figsize=(14,7))
ax = sns.distplot(sim_profits, bins=20, label='original call center simulation re
ax.set_xlabel("Profits",fontsize=16)
ax.set_ylabel("Frequency",fontsize=16)
plt.legend()
```

```
C:\Users\Akshay\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futu
reWarning: `distplot` is a deprecated function and will be removed in a future
version. Please adapt your code to use either `displot` (a figure-level functio
n with similar flexibility) or `histplot` (an axes-level function for histogram
s).
  warnings.warn(msg, FutureWarning)
```

Out[20]:

```
<matplotlib.legend.Legend at 0x5f54460>
```



In [ ]: