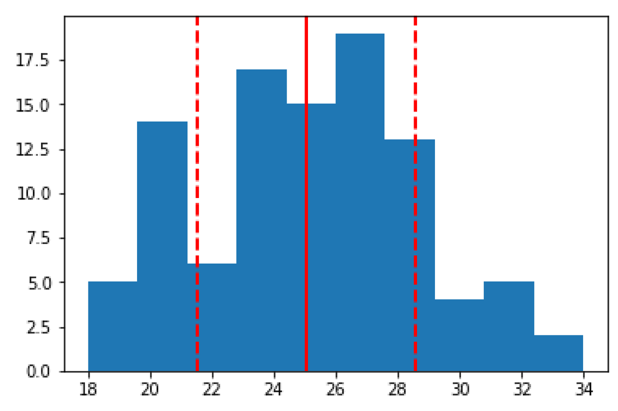
In [1]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

Data Distributions

In [23]:
```python
binomial = np.random.binomial(50, 0.5, 100)

mean = binomial.mean()
print("mean is", mean)

std = binomial.std()
print("standard deviation is", std)

plt.hist(binomial)
plt.axvline(binomial.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(binomial.mean() + binomial.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(binomial.mean() - binomial.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

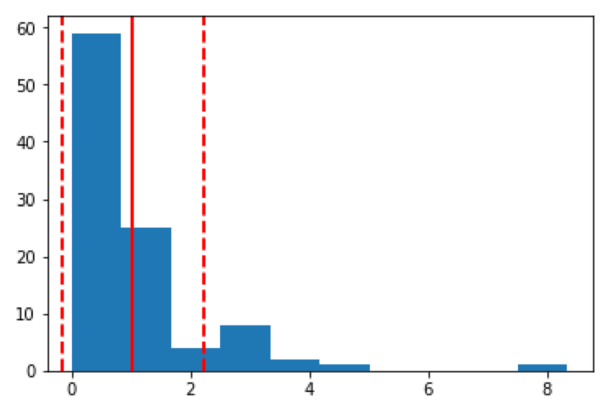
```
mean is 25.06
standard deviation is 3.512321169824878
```



The binomial distribution has a mean that is pretty close to the center of the data so it is still a good measure of central tendency. However, the standard deviations do not cover as much of the data as in a normal distribution so this is not a good measure of variance in the data.

In [25]:
```python
exponential = np.random.exponential(1, 100)

mean = exponential.mean()
print("mean is", mean)

std = exponential.std()
print("standard deviation is", std)

plt.hist(exponential)
plt.axvline(exponential.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(exponential.mean() + exponential.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(exponential.mean() - exponential.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

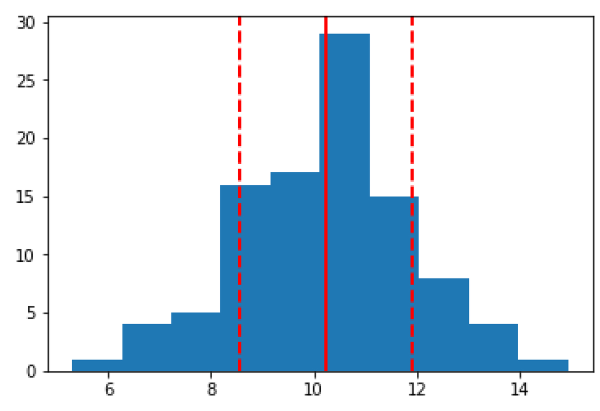
```
mean is 1.0346380605940486
standard deviation is 1.1816636760685464
```



The exponential distribution is skewed to the left so neither the mean nor standard deviations seem to be good measures of central tendency or variance.

In [26]:
```python
logistic = np.random.logistic(10, 1, 100)

mean = logistic.mean()
print("mean is", mean)

std = logistic.std()
print("standard deviation is", std)

plt.hist(logistic)
plt.axvline(logistic.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(logistic.mean() + logistic.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(logistic.mean() - logistic.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

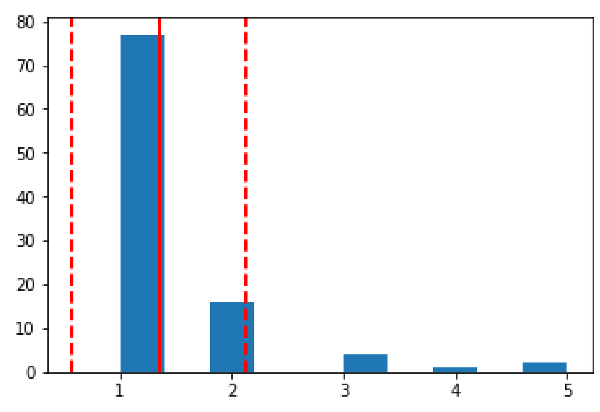
```
mean is 10.234214453730738
standard deviation is 1.6791179621368895
```



The logistic distribution closely resembles a normal distribution, so the mean is a good measure of central tendency and the standard deviation is a good measure of variance.

In [27]:
```python
logseries = np.random.logseries(0.5, 100)

mean = logseries.mean()
print("mean is", mean)

std = logseries.std()
print("standard deviation is", std)

plt.hist(logseries)
plt.axvline(logseries.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(logseries.mean() + logseries.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(logseries.mean() - logseries.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

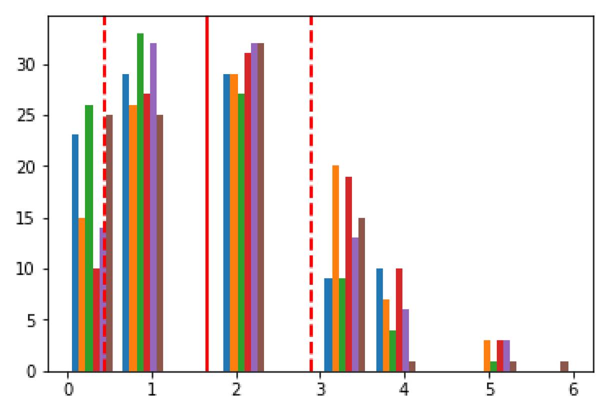
```
mean is 1.35
standard deviation is 0.7794228634059948
```



Neither mean nor standard deviation seem to describe the logseries distribution well.

In [28]:
```python
multinomial = np.random.multinomial(10, [1/6.]*6, 100)

mean = multinomial.mean()
print("mean is", mean)

std = multinomial.std()
print("standard deviation is", std)

plt.hist(multinomial)
plt.axvline(multinomial.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(multinomial.mean() + multinomial.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(multinomial.mean() - multinomial.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

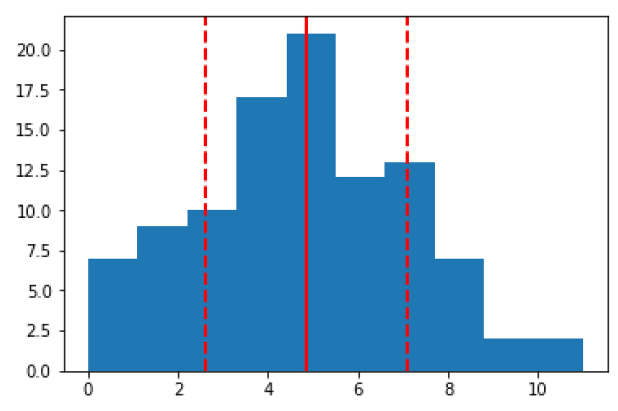
```
mean is 1.6666666666666667
standard deviation is 1.2310790208412925
```



The multinomial distribution appears to be skewed to the left. Both mean and standard deviation appear to be poor descriptive statistics.

In [29]:
```python
poisson = np.random.poisson(5, 100)

mean = poisson.mean()
print("mean is", mean)

std = poisson.std()
print("standard deviation is", std)

plt.hist(poisson)
plt.axvline(poisson.mean(), color='r', linestyle='solid', linewidth=2)
plt.axvline(poisson.mean() + poisson.std(), color='r', linestyle='dashed', linewidth=2)
plt.axvline(poisson.mean() - poisson.std(), color='r', linestyle='dashed', linewidth=2)

plt.show()
```

```
mean is 4.85
standard deviation is 2.228788908802267
```

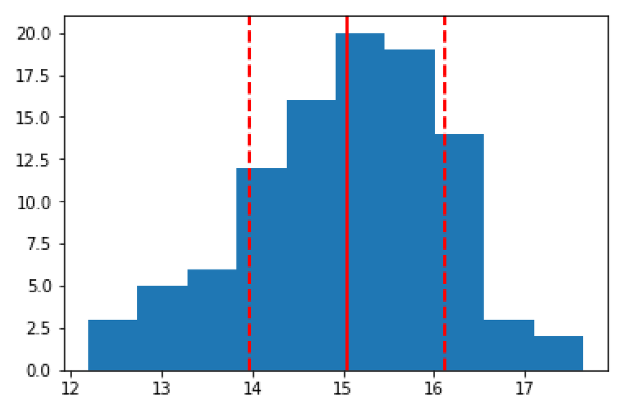


The poisson distribution looks close to a normal distribution so I would say both mean and standard deviation are good measures to describe the data.

For distributions that more closely resemble normal distributions, the mean and standard deviations describe and fit the data much better than distributions that are skewed. Skewed distributions should use other descriptive statistics to best describe central tendency and variance.

In [30]:
```python
norm1 = np.random.normal(5, 0.5, 100)
norm2 = np.random.normal(10, 1, 100)
norm3 = norm1 + norm2

mean = norm3.mean()
print("mean is", mean)

std = norm3.std()
print("standard deviation is", std)

plt.hist(norm3)
plt.axvline(norm3.mean(), color = 'r', linestyle = 'solid', linewidth = 2)
plt.axvline(norm3.mean() + norm3.std(), color = 'r', linestyle = 'dashed', linewidth = 2)
plt.axvline(norm3.mean() - norm3.std(), color = 'r', linestyle = 'dashed', linewidth = 2)

plt.show()
```

```
mean is 15.04286278835214
standard deviation is 1.0773265077013725
```



Adding the two normal distributions seems to have skewed the data but mean still looks like a good descriptor of central tendency and standard deviation looks like a fair measure of variance.