here are two types of Congestion control algorithms, which are as follows −

* Leaky Bucket Algorithm
* Token Bucket Algorithm

Leaky Bucket Algorithm

Let see the working condition of Leaky Bucket Algorithm −

Leaky Bucket Algorithm mainly controls the total amount and the rate of the traffic sent to the network.

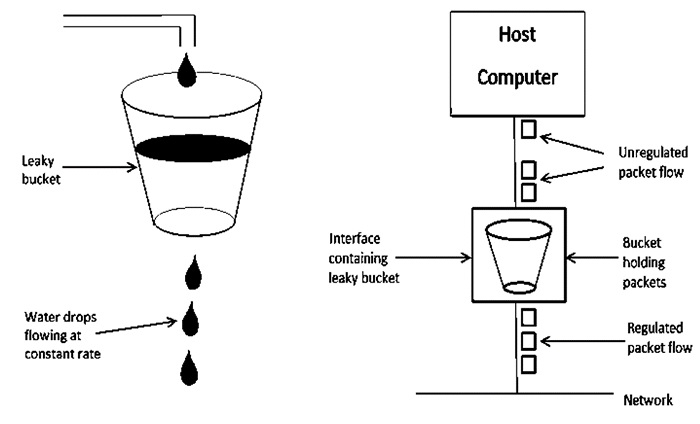
**Step 1** − Let us imagine a bucket with a small hole at the bottom where the rate at which water is poured into the bucket is not constant and can vary but it leaks from the bucket at a constant rate.

**Step 2** − So (up to water is present in the bucket), the rate at which the water leaks does not depend on the rate at which the water is input to the bucket.

**Step 3** − If the bucket is full, additional water that enters into the bucket that spills over the sides and is lost.

**Step 4** − Thus the same concept applied to packets in the network. Consider that data is coming from the source at variable speeds. Suppose that a source sends data at 10 Mbps for 4 seconds. Then there is no data for 3 seconds. The source again transmits data at a rate of 8 Mbps for 2 seconds. Thus, in a time span of 8 seconds, 68 Mb data has been transmitted.

That’s why if a leaky bucket algorithm is used, the data flow would be 8 Mbps for 9 seconds. Thus, the constant flow is maintained.

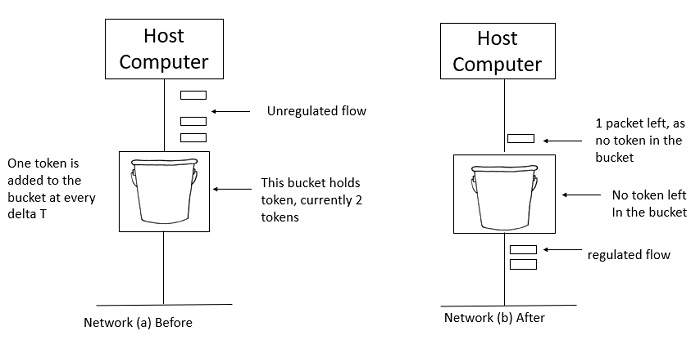


## Token Bucket Algorithm

The leaky bucket algorithm enforces output patterns at the average rate, no matter how busy the traffic is. So, to deal with the more traffic, we need a flexible algorithm so that the data is not lost. One such approach is the token bucket algorithm.

Let us understand this algorithm step wise as given below −

* **Step 1** − In regular intervals tokens are thrown into the bucket f.
* **Step 2** − The bucket has a maximum capacity f.
* **Step 3** − If the packet is ready, then a token is removed from the bucket, and the packet is sent.
* **Step 4** − Suppose, if there is no token in the bucket, the packet cannot be sent.
* In figure (a) the bucket holds two tokens, and three packets are waiting to be sent out of the interface.

****

* In Figure (b) two packets have been sent out by consuming two tokens, and 1 packet is still left.
* When compared to Leaky bucket the token bucket algorithm is less restrictive that means it allows more traffic. The limit of busyness is restricted by the number of tokens available in the bucket at a particular instant of time.
* The implementation of the token bucket algorithm is easy − a variable is used to count the tokens. For every t seconds the counter is incremented and then it is decremented whenever a packet is sent. When the counter reaches zero, no further packet is sent out.

