# Table 11: Performance Metrics – Methods and Measurements

|  |  |  |
| --- | --- | --- |
| Metric Name | Measurement Method | What It Measures |
| Throughput | Measure data processed per unit of time (e.g., MB/s or GB/s). | Speed at which an algorithm can process data. Higher is typically better. |
| Latency | Measure the time taken to complete a single operation (e.g., ms or μs). | Delay to produce an output. Lower is typically better. |
| CPU Utilization | Monitor CPU resources consumed during the operation (%). | The amount of computational resources used. Lower is better for efficiency, but higher might mean faster ops. |
| Memory Footprint | Monitor RAM usage during the cryptographic operation (e.g., KB or MB). | Memory consumed by the algorithm. Lower is better for constrained environments. |
| Energy Consumption | Use power measurement tools to gauge energy use (e.g., mJ or J). | Total energy used during the operation. Important for mobile or battery-operated devices. |
| Key Generation Time | Measure the time to generate cryptographic keys (e.g., ms or μs). | Speed of setting up secure operations. |
| Key Size | Record the length of the cryptographic key (e.g., bits). | Security and storage implications. Typically, longer keys are more secure but might be slower. |
| Cycles per Byte (CPB) | Measure the number of CPU cycles to process a byte of data. | Efficiency of an algorithm, especially in software implementations. |
| Fault Resistance | Introduce faults and observe behavior. | Ability of an algorithm or system to resist fault attacks. |
| Side-Channel Resistance | Test using side-channel analysis techniques. | Resistance to attacks that exploit information leaked during computation (like power consumption patterns). |