Problem Statement

System Responsiveness Guard

Preventing system slowdowns and unresponsiveness caused by resource-hungry processes using a safe, kernel-assisted approach.

Proposed Solution

System Responsiveness Guard is a kernel-assisted mechanism designed to preserve system responsiveness under heavy computational load. The system continuously monitors processlevel scheduling metrics such as CPU utilization, execution time, waiting time, and context switch behaviour within the Linux kernel. Using a clear rule-based detection approach, processes that excessively consume CPU resources over a sustained period are identified.

Once detected, the system applies temporary, non-destructive control actions such as reducing scheduling priority, limiting CPU usage, or adjusting CPU affinity. These actions allow interactive and critical processes to regain fair CPU access while ensuring that the resource-intensive process continues execution at a controlled pace. All interventions are automatically reversed once system stability and responsiveness are restored, ensuring fairness and safety.

Implementation Architecture

The prototype implementation demonstrates the feasibility of kernel-level monitoring and control using a loadable Linux kernel module. A kernel timer periodically triggers the monitoring routine, which scans active processes and evaluates their CPU usage patterns. If a process exceeds predefined thresholds, its scheduling priority is temporarily adjusted using standard kernel APIs. Kernel logs record detection and control actions, enabling easy validation during demonstration. This implementation serves as a proof-of-concept and establishes a foundation for further enhancements.

Impact

System Responsiveness Guard improves overall system usability by preventing any single process from monopolizing CPU resources. By reacting quickly at the kernel level, the system reduces lag, avoids UI freezes, and restores responsive behavior in real time. The solution is especially beneficial in shared and multi-user environments where system responsiveness directly affects productivity.

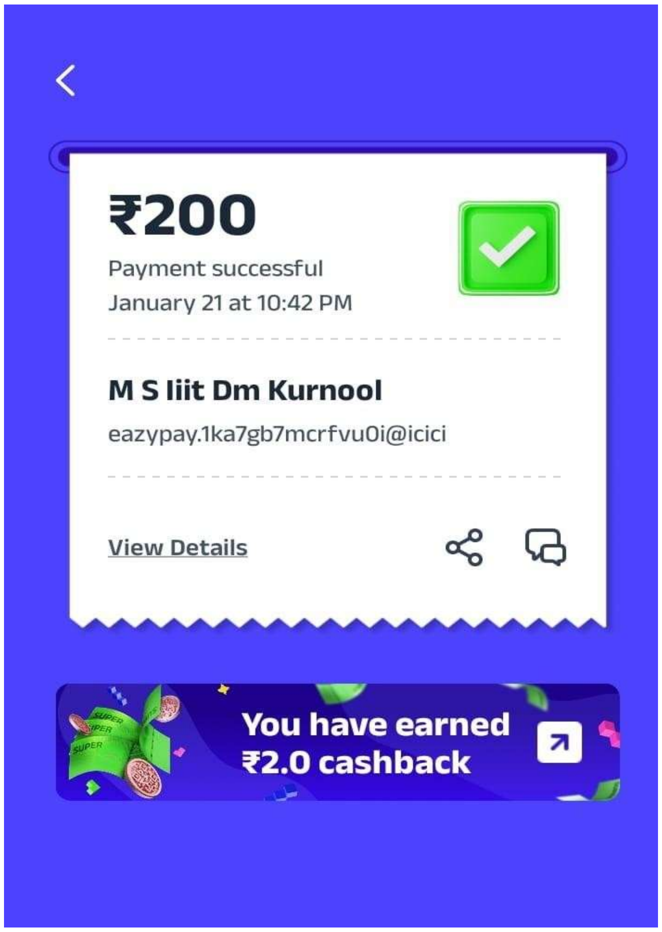
Benefits

The solution maintains system stability by avoiding aggressive actions such as process termination or kernel source modification. Its rule-based design ensures predictable, transparent, and explainable behaviour. The system is lightweight, reversible, compatible with standard Linux kernels, and suitable for real-world deployment as well as academic and hackathon evaluation.

Future Scope and Enhancements

Future enhancements include incorporating additional metrics such as context switch frequency, wait time, and scheduling latency for more accurate detection. Adaptive thresholds can be introduced to handle varying workloads dynamically. User-space visualization and control tools may be developed to improve observability. Advanced mechanisms such as c-groups and CPU affinity can further enhance fine-grained resource management, supported by performance evaluation and stress testing in real-world scenarios.

Payment Proof:



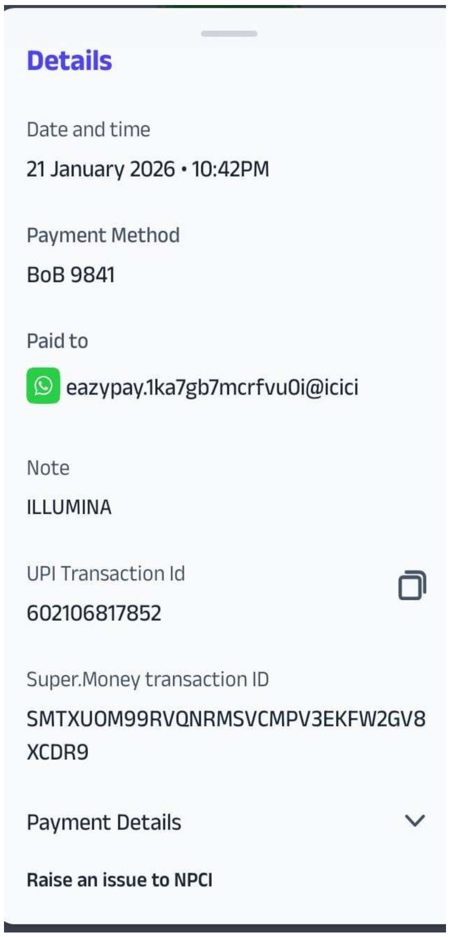
₹200

Payment successful

January 21 at 10:42 PM

M S Iiit Dm Kurnool

eazypay.1ka7gb7mcrfvu0i@icici



Details

Date and time

21 January 2026 • 10:42PM

Payment Method

BoB 9841

Paid to

eazypay.1ka7gb7mcrfvu0i@icici

Note

ILLUMINA

UPI Transaction Id

602106817852

Super.Money transaction ID

SMTXUOM99RVQNRMSVCMPV3EKFW2GV8XCDR9