TABLE VII: Identifying and Addressing Real-Time Faults: Understanding Root Causes and remediation

Failure type	Root cause	Remediation steps
Operational Error	Connection Limit Reached	1- Increase Connection Limit 2- Identify Resource-Intensive Queries 3- Implement Retry Logic 4- Monitor and Scale 5- Database Connection Pooling
Connection refused	MySQL Server Unreachable	1- Check MySQL Server Status 2- Check Network Connectivity 3- Check MySQL Server Configuration 4- Check MySQL Server Port 5- Check Firewall and Security Groups 6- Check DNS Resolution
System stuck	High memory usage	Check 1- memory usage if high decrease the number of replicas by one
Process crach	Incorrect deallocation	1- Restart the pods to solve the process crash 2- Solve incorrect deallocation
Login failure	Network interruption	1- solve network interruption
Access denied	1- Misconfiguration(service refused)	1- Solve misconfiguration by deleting and recreating the service
DNS errors	Incorrect DNS configurations and CoreDNS Issues	1- Check CoreDNS Pods and Verify Pod DNS Configuration
DNS faults	Mistakes in specifying DNS policies and Kube-DNS or CoreDNS Problem	1- Ensure proper configuration of CoreDNS, the default DNS server
Node I/O stress	The node is running low on resources and impacting the node's overall performance	1- Implement resource management techniques: such as Quality of Service (QoS) classes, node affinity/anti-affinity, resource limits/requests to optimize and control I/O usage on nodes
Pod API latency	Node Overload: lead to latency across all Pods on that node	1- Optimize the number of containers per pod and utilize resource requests and limits effectively
Overrides header values of API requests	Ingress Resource Configuration	1- Involve "MutatingAdmissionWebhook" to intercept requests and modify headers
Node memory hog	memory leak and Large or unoptimized Workloads	1- Involve identifying and addressing memory-intensive pods optimizing resource requests and limits, implementing Horizontal Pod Autoscaling
Resource overload	Insufficient Resource Requests and Limits and Pod Autoscaling	1- Adjust the number of pod replicas based on resource metrics (CPU and memory usage)