A Security Engineering Report

Impact of Modern Security Software on InterSystems IRIS



RELEASE:
SEPTEMBER 2026

InterSystems®
Creative data technology

With proper tuning and validation, endpoint protection can be safely integrated into InterSystems IRIS deployment, achieving robust security without significant performance degradations

Table of Contents

Executive Summary	3
The Need for Endpoint Protection on Operational Databases	3
Test Environment and Methodology	4
Performance Results	5
CPU and ObjectScript Language Execution (%BENCHLANG)	6
Journal Performance	7
Dejournaling	8
Journal Transfer	10
Process Creation and Concurrency (Jobs Tests)	12
Local Jobs	12
Remote (ECP) Jobs	13
Saturate Microbenchmarks	14
Hardware-Level Synchronization/Timed System Metrics (TSM)	15
Analysis and Interpretation	17
Recommendations	18
Conclusion	19

This report evaluates the impact of modern endpoint protection software, specifically Palo Alto Cortex XDR and CrowdStrike Falcon, on the performance of **InterSystems IRIS**® databases running on RHEL 9.5 and Ubuntu 24.04. Benchmarking tests covering journaling, process creation, and memory coordination show that core database operations are largely unaffected by endpoint protection in most scenarios. However, journaling activities exhibit modest, environment-specific performance variability, with the most notable being an 8.3% slowdown in the Ubuntu XDR configuration. The results suggest that with proper tuning and validation—particularly for journaling paths—endpoint protection can be safely integrated into InterSystems IRIS deployments. The study concludes that robust security can be achieved without significant performance degradation, supporting the use of endpoint protection as a standard practice in use of InterSystems IRIS.

Executive Summary

As cyber threats grow more frequent and sophisticated, organizations operating mission-critical databases face mounting pressure to deploy robust Endpoint Detection and Response (EDR) or Extended Detection and Response (XDR) software. This is particularly true in healthcare and other industries that hold valuable data in databases, where industry reports have shown that 80% of ransomware attacks involve critical infrastructure such as operational databases. Databases are often compromised within 30 minutes of an initial breach, underscoring the need for modern endpoint protection software. However, database platforms such as InterSystems IRIS are highly optimized systems that depend on predictable latency, efficient I/O operations, and scalable concurrency. Introducing additional layers of security, especially those operating at the kernel and file system level, raises legitimate concerns about performance degradation.

This paper presents a comprehensive analysis of the performance impact on InterSystems IRIS database workloads of two widely adopted endpoint protection solutions: Palo Alto Cortex XDR and CrowdStrike Falcon. Testing was conducted across both Red Hat Enterprise Linux (RHEL) 9.5 and Ubuntu 24.04 environments using a suite of benchmarks designed to simulate real-world database activity, including journaling, process scaling, and memory contention.

Overall, the results show that modern endpoint protection can be deployed on InterSystems IRIS without disrupting normal database operations. Core processing, concurrency, and day-to-day workloads continue to run as expected, with no noticeable impact in most scenarios. Some variability may occur in journaling-related activities depending on operating system and agent configuration, but these effects are limited and manageable in practice. **These findings support the conclusion that endpoint protection can be safely deployed on InterSystems IRIS systems, provided that implementation is accompanied by appropriate tuning and validation.**

The Need for Endpoint Protection on Operational Databases

Modern database systems are no longer deployed in isolated security zones within a data center; they are integral components of interconnected, distributed architectures and are increasingly exposed to sophisticated attack vectors. Across healthcare, financial services, and supply chain environments, databases serve as repositories for highly sensitive and mission-critical information that underpins patient care, financial transactions, and operational logistics. As a result, they are prime targets for ransomware, data exfiltration, and insider threats. Attackers frequently leverage lateral movement techniques to navigate enterprise environments and gain access to systems running operational databases. This underscores the need for behavioral and anomaly-based detection.

Traditional antivirus solutions, however, relied heavily on signature-based detection and performed file scanning on every read and write operation. These approaches introduced significant performance overhead, particularly for high-throughput systems such as databases. To mitigate this impact, software vendors often recommended allowlisting database

directories and executables, reducing scanning overhead but also limiting security visibility and increasing exposure to attack. InterSystems provides guidance for administrators considering these approaches in the InterSystems IRIS documentation: **https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=GSA_thirdparty.**

Endpoint protection platforms serve as a critical defense layer by providing capabilities such as behavioral analysis, exploit prevention, file integrity monitoring, and real-time threat detection. These tools operate continuously, inspecting file access patterns, monitoring process behavior, and intercepting potentially malicious activity to prevent systems from being compromised.

Rather than relying primarily on file-based scanning, modern endpoint protection platforms leverage behavioral analysis, process context, and telemetry correlation to detect malicious activity. This shift significantly reduces the performance impact on high-throughput applications such as database systems while improving detection of advanced threats.

Despite these improvements, there is a longstanding concern within database operations teams that security tools may still interfere with performance-sensitive workloads. InterSystems IRIS databases rely heavily on high-throughput journaling, efficient process spawning, and low-latency memory operations. The introduction of monitoring mechanisms such as kernel hooks, process inspection, and telemetry collection has the potential to impact these critical paths.

This tension between security and performance requires real-world testing. Rather than relying on assumptions or anecdotal evidence, this study quantifies the real-world performance impact of endpoint protection in a controlled test environment by measuring InterSystems IRIS workloads and operating system behaviors with and without endpoint protection enabled.

Test Environment and Methodology

The testing environment was designed to reflect a high-performance production deployment while maintaining isolation for accurate measurement. Two systems were used. The first was powered by dual Intel Xeon Platinum processors with 60 cores per socket, and the second, powered by dual AMD EPYC processors with 32 cores per socket. These systems were connected via a high-speed 100Gb network and placed within an isolated lab environment to eliminate external variability.

Both systems were configured with standardized InterSystems IRIS settings to ensure consistency across test runs. Parameters such as global buffers, memory heap allocation, and journaling configuration were held constant. This eliminated configuration drift as a variable and ensured that any observed differences could be attributed to the presence or absence of endpoint protection. Testing was conducted between June and July 2025 using production versions of CrowdStrike Falcon and Palo Alto Cortex XDR agents available at that time.

The testing included:

- Two operating systems: RHEL 9.5 and Ubuntu 24.04
- Three configurations:
  - Baseline (no endpoint protection)
  - Palo Alto Cortex XDR
  - CrowdStrike Falcon
- Multiple benchmarks runs per configuration to ensure consistency and reduce transient system noise

Although operating system differences are included in the analysis, they are not the focus of this study. The primary goal is to isolate the impact of endpoint protection, and results are interpreted accordingly. Observed OS-level differences are treated as environmental context rather than primary findings. A full statistical analysis is provided as a supplemental document to this paper.

The benchmark suite was intentionally diverse. The **%BENCHLANG** test focused on ObjectScript execution performance, while journaling tests evaluated disk and replication behavior. Jobs tests measured process scalability, and the saturate microbenchmarks simulated internal InterSystems IRIS engine operations such as locking and block contention. Finally, the TSM benchmark provided insight into low-level hardware synchronization behavior.

Performance Results

This section presents the results of the benchmark suite used to evaluate the performance impact of endpoint protection on InterSystems IRIS. Figure 1 summarizes the layered approach to the testing.

As with all performance benchmarking in high-performance systems, some degree of variability is expected between runs, as caused by factors such as operating system scheduling, caching behavior, and background system activity. The results presented here reflect consistent patterns observed across multiple runs, and small differences between configurations should be interpreted within the context of normal system variability rather than as definitive performance changes. The tests are organized to reflect different layers of system behavior, beginning with CPU-bound ObjectScript execution, followed by I/O-intensive journaling operations and process creation under increasing levels of concurrency. The analysis examines high-concurrency database engine behavior using the **^saturate** benchmarks then examines low-level system primitives through Timed System Metrics (TSM). This structured approach allows us to identify where endpoint protection introduces measurable overhead, if any, across the software and hardware stack. This progression mirrors how database workloads interact with the system, enabling clearer attribution of any observed performance effects.

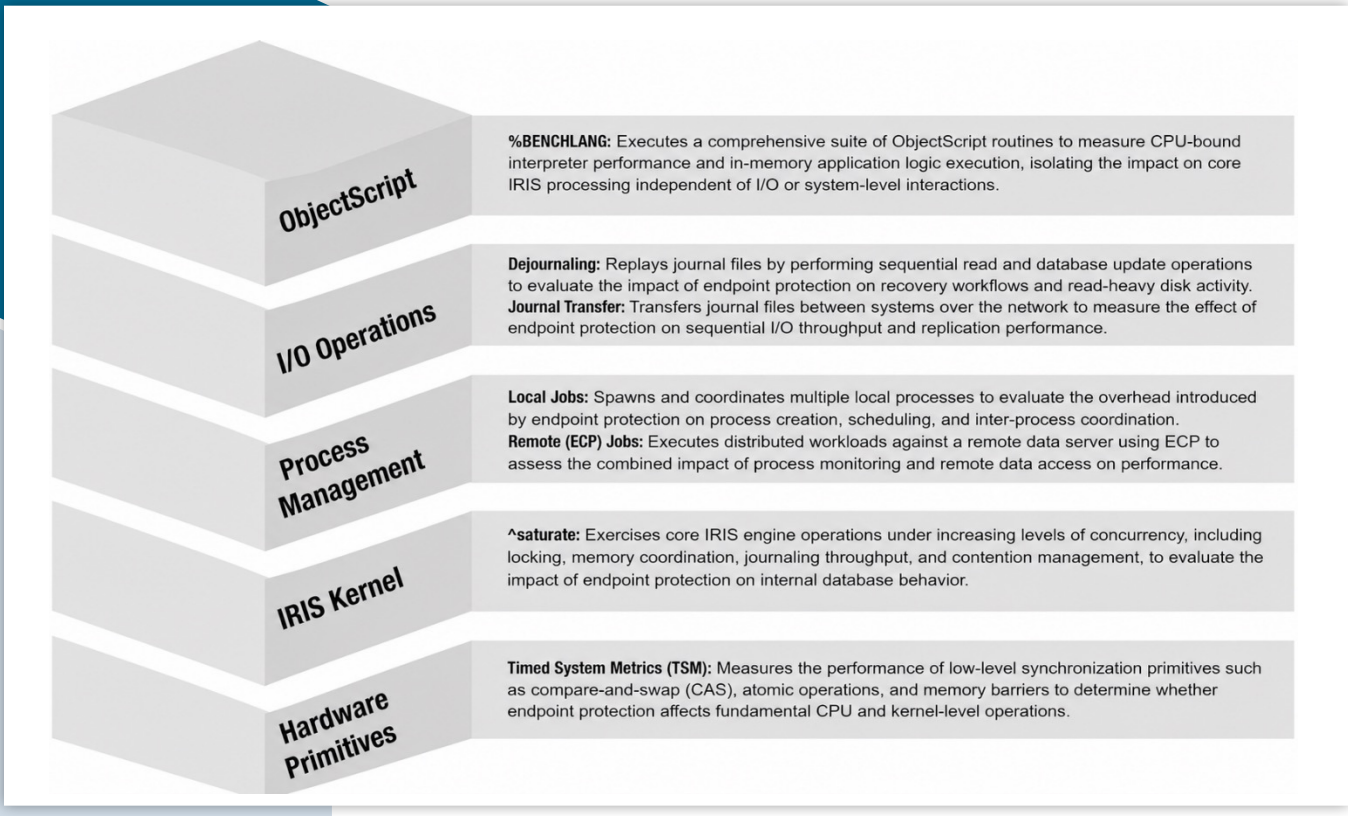


Figure 1: Layered Approach to Performance Testing

CPU and ObjectScript Language Execution (%BENCHLANG)

One of the core components of InterSystems IRIS is the ObjectScript virtual machine, which underpins application execution and memory management. Because InterSystems IRIS is designed to maximize in-memory processing, this benchmark focuses on CPU-bound behavior with minimal reliance on disk or operating system–level interaction.

Testing was performed across RHEL 9.5 and Ubuntu 24.04, comparing baseline systems with deployments of Cortex XDR and CrowdStrike Falcon. COSMark Score results were consistent across all configurations. Endpoint protection configurations may appear slightly faster than baseline. This does not indicate that endpoint protection improves performance, but reflects normal system variability in high-performance environments.

The violin plot in Figure 2 shows substantial overlap between all configurations, for both RHEL 9.5 and Ubuntu 24.04, indicating no meaningful difference in performance. The descriptive statistics for this test are shown in Table 1.

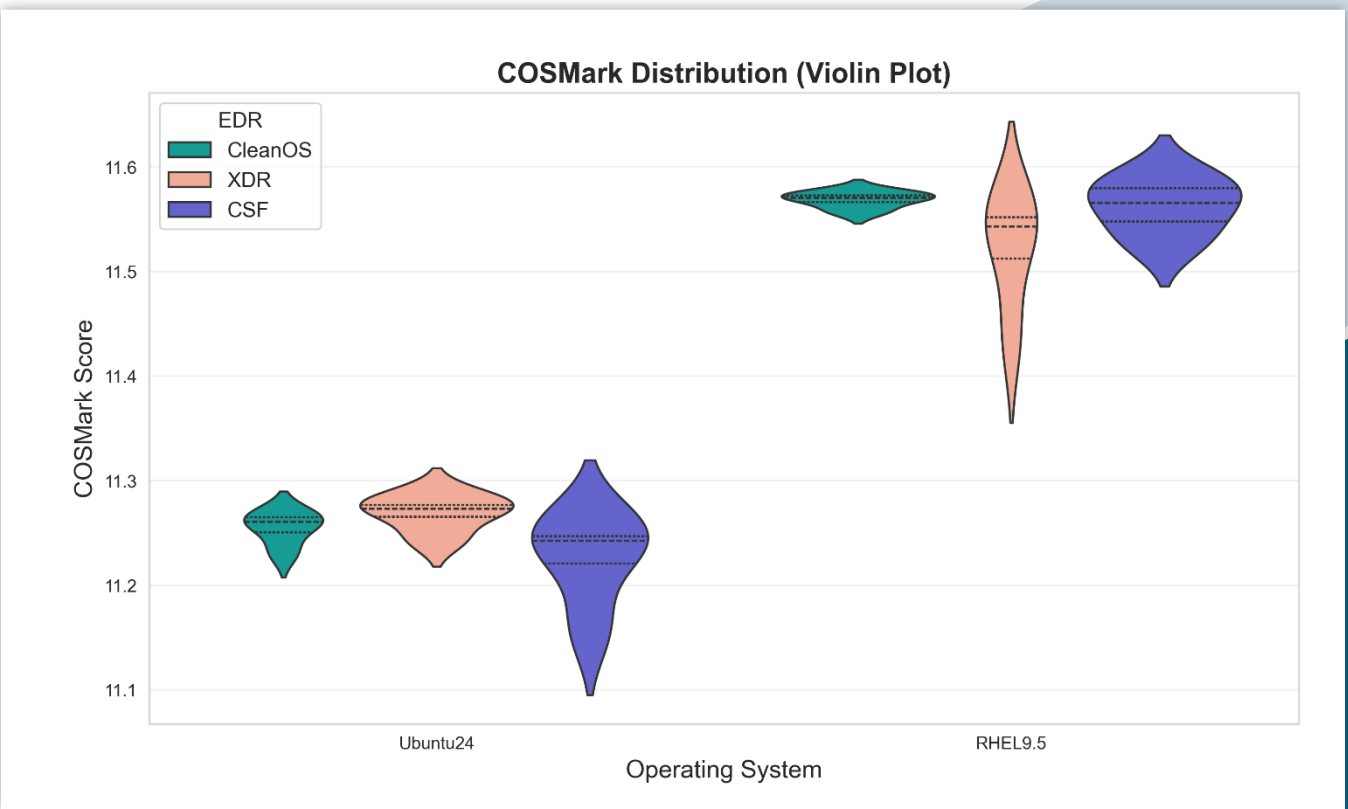


Figure 2: Violin Plot of COSMark Distribution Across Test Groups

OS	EDR	Mean	Std Deviation
RHEL 9.5	CSF	11.561775	0.027227
	CleanOS	11.568525	0.007801
	XDR	11.521100	0.055371
Ubuntu24	CSF	11.225025	0.043290
	CleanOS	11.254700	0.015903
	XDR	11.269075	0.017483

Table 1: Descriptive Statistics of COSMark Score Distribution Tests

Overall, ObjectScript execution performance remains stable with endpoint protection enabled.

⚠ **In our evaluation, endpoint protection has no meaningful impact on CPU-bound ObjectScript execution.**

Journal Performance

Journaling is a critical component of InterSystems IRIS, the foundation for durability, recovery, and replication. With Journaling, every operation that modifies data is recorded sequentially in a journal file, creating a persistent log of changes that can be used to restore or synchronize databases. This write-ahead logging mechanism ensures that, in the event of a system failure, transactions can be replayed to maintain consistency and prevent data

loss. Journaling is also fundamental to mirroring and disaster recovery configurations, where journal records are transmitted and reapplied on secondary systems to keep them in sync with the primary. Journaling involves high-frequency, latency-sensitive write operations. Because of this, it represents one of the most performance-critical I/O paths in InterSystems IRIS. Any added overhead such as delays introduced by file inspection or interception can impact transaction throughput, recovery time, and overall system responsiveness.

To evaluate the impact of endpoint protection on journaling operations, the testing focuses on dejournaling and journal transfer benchmarks. Dejournaling measures the performance of replaying journaled operations back into the database, while journal transfer evaluates the efficiency of transmitting journal files between systems in a mirrored configuration. The act of writing journal files themselves was not explicitly examined in this section, as journaling is tightly integrated into normal database operation and is already reflected in overall system performance benchmarks.

Together, these tests provide a representative view of how endpoint protection may affect both local recovery workflows and distributed replication scenarios.

Dejournaling

Dejournaling is the process of reading journal files and replaying the recorded database operations to apply those changes to a database. It is a critical component of IRIS mirroring, supporting both synchronous and asynchronous mirror members, and directly impacts failover behavior, recovery time objectives (RTO), and how quickly secondary systems can catch up with the primary. Unlike journaling, which is write-intensive and part of the transaction commit path, dejournaling is primarily a sequential read operation combined with a series of database updates. Read operations are generally less sensitive to the types of interception or inspection performed by modern endpoint protection and XDR platforms. As the dejournaling workload is dominated by sequential reads of journal files, a significant performance impact from endpoint protection is not expected during dejournaling. Because dejournaling underpins mirror catch-up and failover readiness, any performance impact in this area has direct implications for high availability and disaster recovery operations.

The dejournaling test used in our research simulated recovery scenarios by processing 20 journal files, each 4GB in size, with updates applied across 20 databases. This workload is representative of recovery and replay operations, placing sustained demand on I/O, coordination, and system-level scheduling. Results were consistent across operating systems and endpoint protection configurations, with only minor variation observed between runs. Differences between configurations were small and aligned with expected system variability for this type of workload. Table 2 and Figure 3 shows these results on the next page.

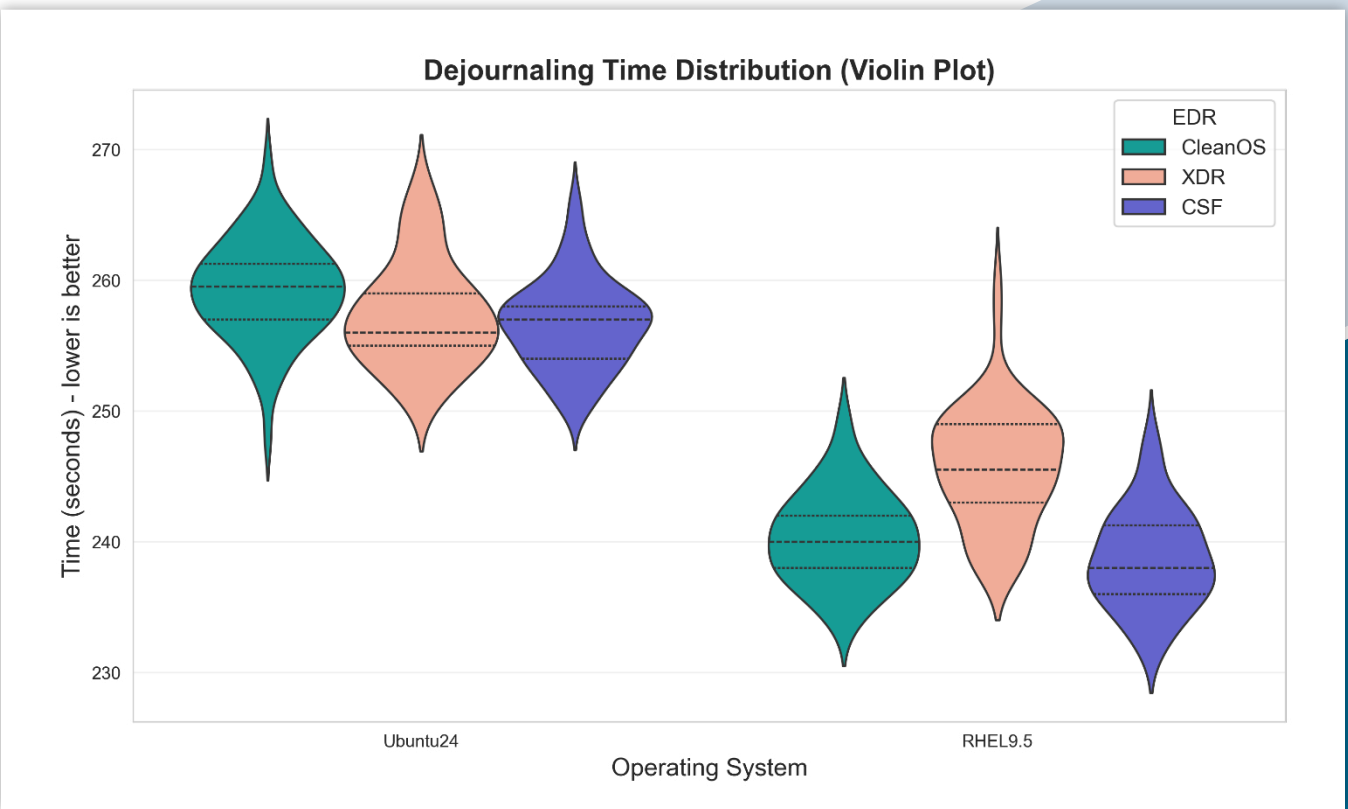


Figure 3: Dejournaling Time Distribution

OS	EDR	Mean	Std Deviation
RHEL9.5	CSF	238.650000	3.745425
	CleanOS	240.300000	3.475530
	XDR	245.700000	4.540962
Ubuntu24	CSF	256.633333	3.404716
	CleanOS	259.266667	3.781609
	XDR	257.433333	4.048698

Table 2: Descriptive Statistics for Dejournaling Time Tests

While EDR introduces statistically significant differences in dejournaling performance, the magnitude of these effects is small and varies by operating system. Within individual platforms, differences between endpoint protection configurations are not consistent, indicating that the observed impact is modest and environment-dependent.

Overall, these results reinforce that endpoint protection operates with minimal overhead during recovery-oriented operations such as dejournaling, with performance driven primarily by underlying system characteristics rather than endpoint protection presence.

⚠ **In our evaluation, endpoint protection has a minor impact on dejournaling performance.**

Journal Transfer

Journal transfer refers to the process of transmitting journal files from a primary InterSystems IRIS system to a secondary system where they can be applied to maintain data consistency. This operation is both a network- and I/O-driven workload, involving the sequential reading of journal files on the source system and their transmission over the network to the remote system. Unlike local database operations, journal transfer depends on sustained throughput across both disk and network interfaces. The test environment utilized a dedicated 100Gb network connection between systems, effectively eliminating network bandwidth as a limiting factor. This ensures that observed performance differences are driven by system behavior and endpoint protection overhead rather than network constraints.

Because this activity is largely sequential and does not involve intensive file-by-file inspection or complex process interactions, based on how endpoint protection operates, we expect minimal impact on journal transfer performance. This is due to their reduced reliance on inline file scanning and greater emphasis on behavioral and telemetry-based analysis. As a result, journal transfer serves as a useful benchmark for validating that replication and data synchronization workflows remain efficient under endpoint protection.

The journal transfer test evaluated replication performance by transferring journal files between systems using the InterSystems IRIS mirror agent. Results were generally consistent with baseline expectations across most configurations, indicating that endpoint protection does not introduce meaningful overhead in typical environments.

On RHEL, both endpoint protection solutions behaved similarly to baseline, with only minor variation observed. On Ubuntu, results were slightly more variable, with one configuration showing a more noticeable performance degradation of 8.3%. This behavior appears to be environment-specific rather than a consistent effect of endpoint protection.

Overall, the findings suggest that journal transfer performance remains stable in most cases, with variability driven more by operating system and agent interaction than by endpoint protection alone.

Table 3 and Figure 4 shows these results.

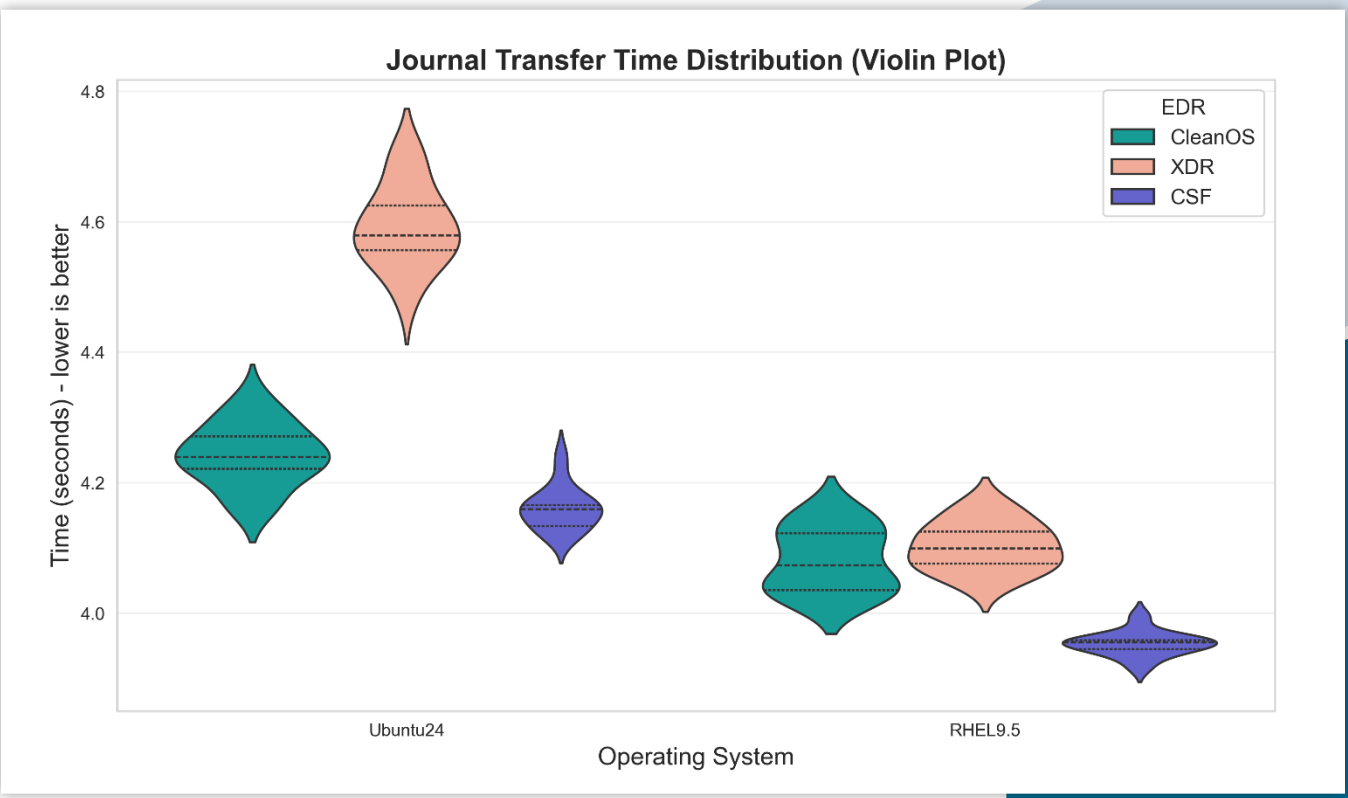


Figure 4: Journal Transfer Time Distribution

OS	EDR	Mean	Std Deviation
RHEL9.5			
	CSF	3.953066	0.018659
	CleanOS	4.081185	0.048410
	XDR	4.102520	0.036867
Ubuntu24			
	CSF	4.160092	0.033107
	CleanOS	4.241961	0.046905
	XDR	4.594100	0.062156

Table 3: Descriptive Statistics for Journal Transfer Times

While the degradation observed on Ubuntu is notable, it is specific to the Ubuntu/XDR configuration. CSF does not show the same behavior, indicating that the effect is tied to a specific platform and environment rather than endpoint protection in general and additional tuning may mitigate the effect. The absence of similar behavior on RHEL suggests that differences in operating system implementation (such as file system behavior, kernel I/O handling, or the way endpoint protection agents integrate with the OS) likely influence performance outcomes.

Overall, journaling performance may vary depending on environment, particularly based on operating system and endpoint agent behavior. While most configurations show stable performance, isolated cases highlight the importance of validation in real-world deployments.

⚠ **In our evaluation, endpoint protection has minor impact on journal transfer, but environment-specific validation is recommended.**

Process Creation and Concurrency (Jobs Tests)

Process creation is the mechanism by which the operating system spawns new jobs to perform tasks in parallel. In InterSystems IRIS, scalability is closely tied to how efficiently these processes can be created and managed, as the platform relies on large numbers of concurrent processes to handle application workloads. The jobs tests were designed to evaluate how effectively the system can spawn and coordinate these processes under increasing levels of concurrency.

This area is relevant when evaluating endpoint protection, as modern endpoint protection platforms monitor process creation, execution, and inter-process behavior as part of their behavioral detection capabilities. Process-intensive workloads are often expected to show some performance impact due to process monitoring. However, in this study, no measurable effect was observed. As a result, process-heavy workloads provide a useful validation that endpoint protection does not introduce overhead in real-world InterSystems IRIS process scaling scenarios.

Local Jobs

Results were consistent across all configurations, with no observable differences attributable to endpoint protection. After accounting for process scaling using a logarithmic model, performance is driven almost entirely by parent process count, with endpoint protection contributing no meaningful effect.

Performance was driven primarily by concurrency and process orchestration characteristics rather than endpoint protection. Variability observed at lower process counts occurred under light system load and reflects normal system behavior; under higher concurrency levels representative of real workloads, performance remained stable with no observable impact from endpoint protection.

Differences between operating systems were minimal, indicating that platform and agent implementation do not materially influence performance in this test. Table 4 and Figure 5 on the next page shows the results of the local jobs tests.

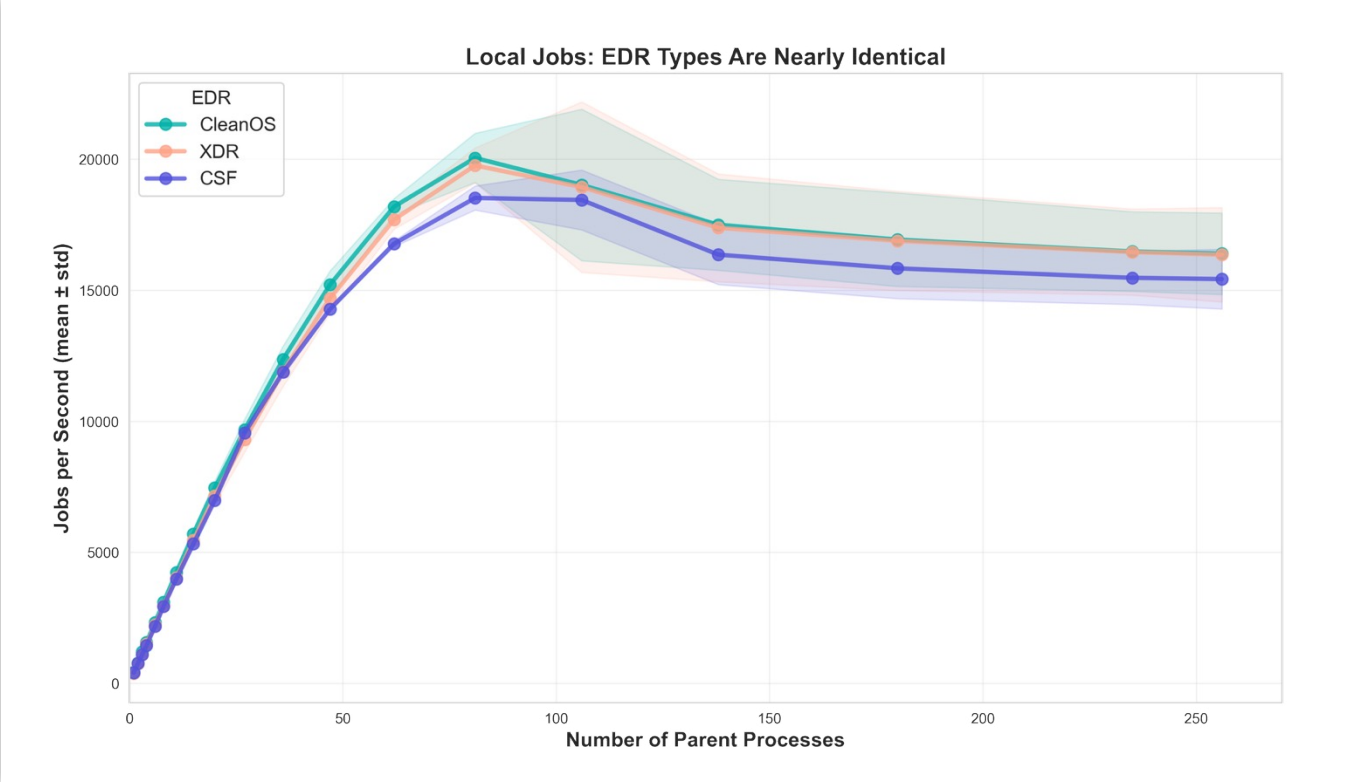


Figure 5: Local Jobs Test Results

OS	EDR	Mean	Std Deviation
RHEL9.5			
	CSF	9568.947368	7102.702040
	CleanOS	10227.578947	7737.568512
	XDR	10051.473684	7742.411614
Ubuntu24			
	CSF	9131.105263	6617.946198
	CleanOS	9623.894737	6907.681191
	XDR	9412.526316	6820.962667

Table 4: Descriptive Statistics for Local Jobs Tests

Overall, performance remains stable for process-intensive workloads with endpoint protection enabled.

⚠ **In our evaluation, endpoint protection has no meaningful impact on local process creation and concurrency under typical high-concurrency workloads, where system performance is most critical.**

Remote (ECP) Jobs

The Remote ECP jobs test evaluates distributed workloads where processes execute against data hosted on a remote system using the Enterprise Cache Protocol (ECP) feature of InterSystems IRIS. This combines process creation with network-based data access, introducing additional system coordination compared to local workloads.

Results were broadly consistent with the local jobs test, showing no observable differences attributable to endpoint protection.

Despite the added complexity of remote data access, performance remained driven by workload design and system characteristics rather than endpoint protection. The high-speed network configuration ensured that results reflect system behavior rather than network limitations.

The results are shown in Table 5 and Figure 6:

Figure 6: Remote Jobs Test Results

OS	EDR	Mean	Std Deviation
RHEL9.5			
	CSF	7641.000000	5240.380181
	CleanOS	8278.684211	5768.104031
	XDR	7882.947368	5551.131861
Ubuntu24			
	CSF	6469.157895	4741.060081
	CleanOS	6794.578947	5198.616753
	XDR	6810.052632	5133.573409

Table 5: Descriptive Statistics for Remote Jobs Tests

Overall, distributed InterSystems IRIS workloads perform as expected with endpoint protection enabled.

⚠ **In our evaluation, endpoint protection has no meaningful impact on remote (ECP) process execution and concurrency under typical high-concurrency workloads.**

Saturate Microbenchmarks

The **^saturate** benchmark evaluates the behavior of core InterSystems IRIS engine operations under high concurrency, including locking, memory coordination, journaling throughput, and contention for shared resources. These tests simulate heavy system load and show how the database engine performs under stress.

Across all workloads and concurrency levels, performance remained consistent between baseline and endpoint protection configurations. While minor variation was observed in some high-concurrency scenarios, these differences were not consistent and reflect normal system behavior. One notable exception was observed in lock contention scenarios, where variability was slightly more pronounced, indicating that highly contended synchronization paths may be more sensitive to environmental factors.

Performance scaled as expected with increasing process counts, indicating that endpoint protection does not interfere with InterSystems IRIS engine scalability or coordination mechanisms. The results across the process range is shown in Table 6 and Figure 7 on the following page.

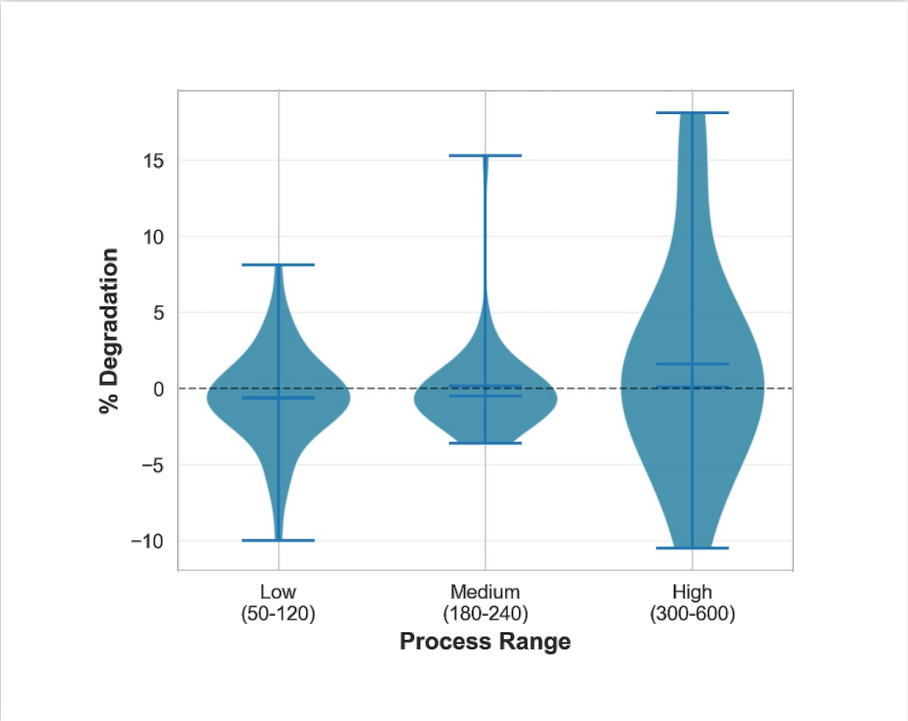


Figure 7: Saturate Results by Process Count

EDR	Mean	Std Deviation
CSF	-0.291	3.514
XDR	0.341	5.065

Table 6: Descriptive Statistics of Overall Degradation for Endpoint Protection Software

Overall, core InterSystems IRIS engine behavior remains stable under load with endpoint protection enabled. While lock contention scenarios showed slightly higher variability, this did not translate into consistent degradation.

⚠ **In our evaluation, endpoint protection has no meaningful impact on high-concurrency database engine operations, including lock contention workloads.**

Hardware-Level Synchronization/Timed System Metrics (TSM)

The Timed System Metrics (TSM) benchmark evaluates low-level synchronization primitives that underpin concurrent processing in InterSystems IRIS. These include atomic operations, memory coordination, and CPU-level synchronization mechanisms that form the foundation for higher-level database behavior.

Results showed natural variability between runs, regardless of whether endpoint protection was enabled. Differences between baseline and protected systems fell within expected system variability, with no observable pattern indicating performance degradation.

These findings demonstrate that endpoint protection does not interfere with low-level system operations or CPU-level behavior.

The results are shown in Figure 8 below and the variability of the time differences across all process counts is shown in Figure 9. The overall descriptive statistics are shown in Table 7.

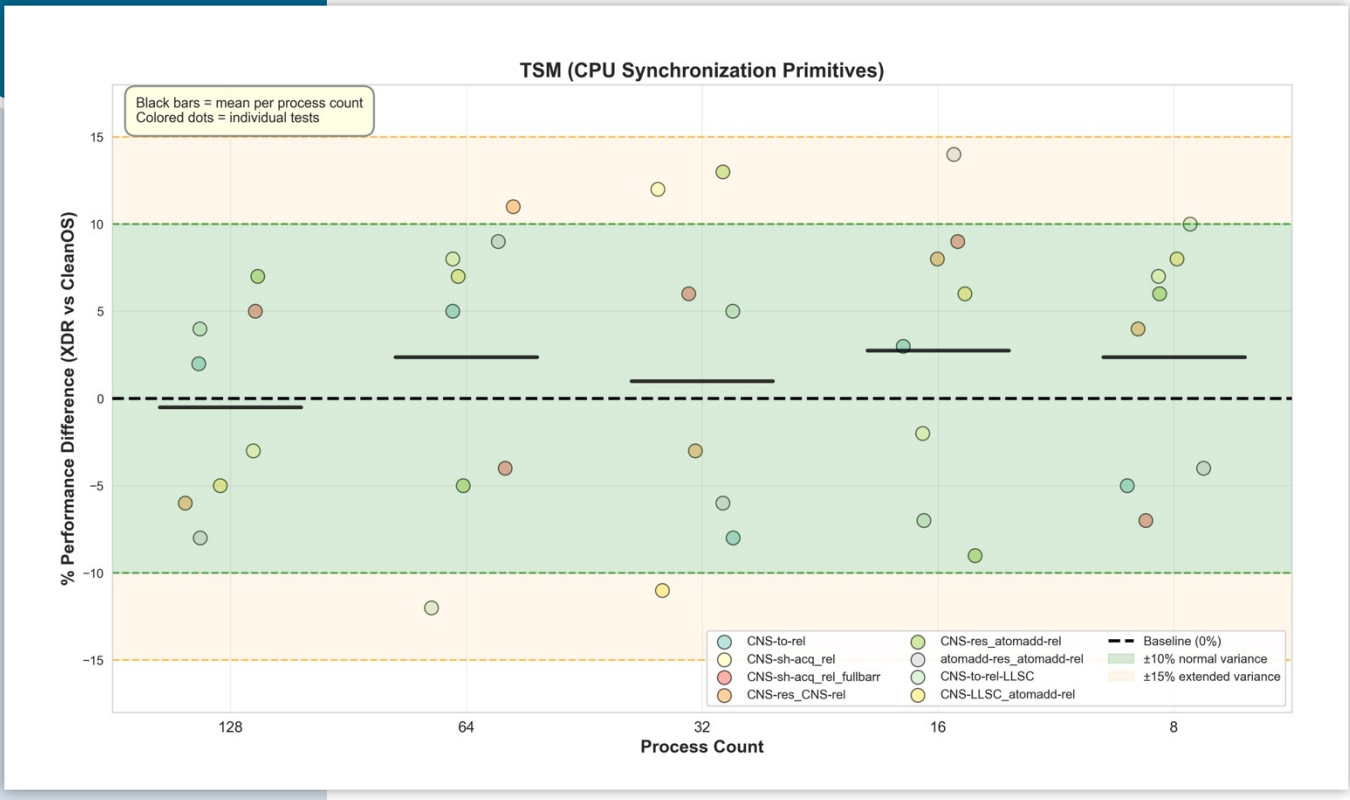


Figure 8: TSM Performance Degradation Across Processes

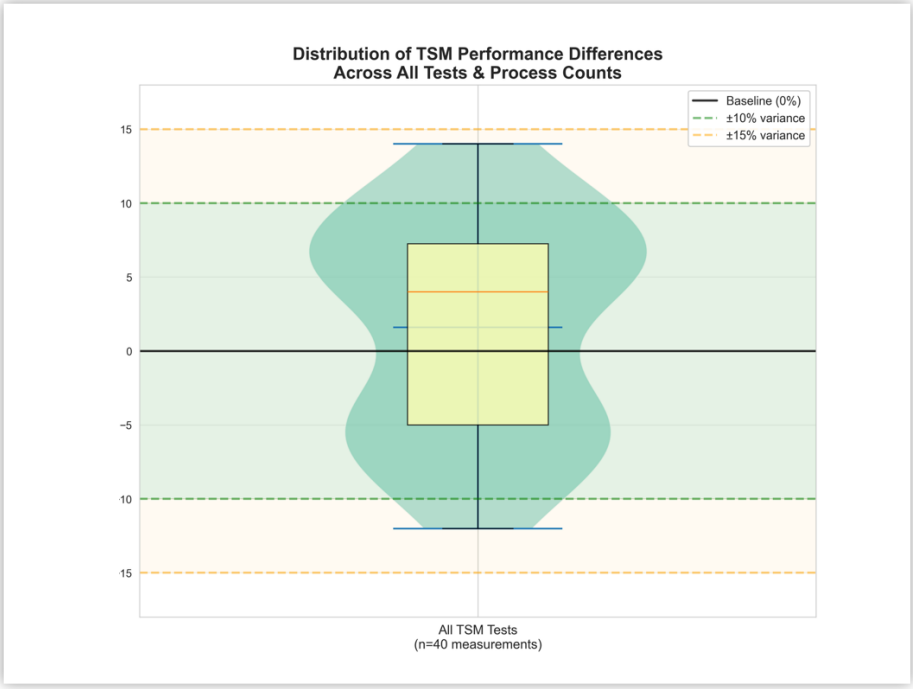


Figure 9: Distribution of TSM Performance Differences

EDR	Mean	Std Deviation
CSF	0.655	3.710826
XDR	-0.255	4.197188

Table 7: Descriptive Statistics for Overall Differences of TSM Tests by Endpoint Protection Software

Overall, performance at the lowest levels of the system stack remains stable with endpoint protection enabled.

⚠ **In our evaluation, endpoint protection has no meaningful impact on low-level CPU and synchronization operations.**

Analysis and Interpretation

The results of this study reveal a clear pattern. For most workloads, the negligible impact of endpoint protection translates directly into stable SLA adherence, predictable response times, and no disruption to application throughput. In healthcare and other mission-critical environments, this means clinical workflows, transaction processing, and real-time decision support systems continue to operate without degradation.

Where moderate overhead was observed in journaling-related activities, the practical implications are limited but important to understand. Small increases in journaling latency may influence failover times, recovery point objectives (RPO), and replication lag in mirrored environments. However, the magnitude of these effects remains within acceptable operational thresholds for most deployments, particularly when systems are properly tuned.

Overall, the results show a clear pattern: modern endpoint protection platforms have minimal impact on typical operational database functionality but can introduce moderate overhead in journaling activity. This distinction is important for understanding real-world implications.

CPU-bound operations and internal mechanics remain stable because they do not trigger the monitoring mechanisms of endpoint protection tools. Figure 9 provides a consolidated view of performance across all test layers. Figure 9 summarizes the results of our testing.

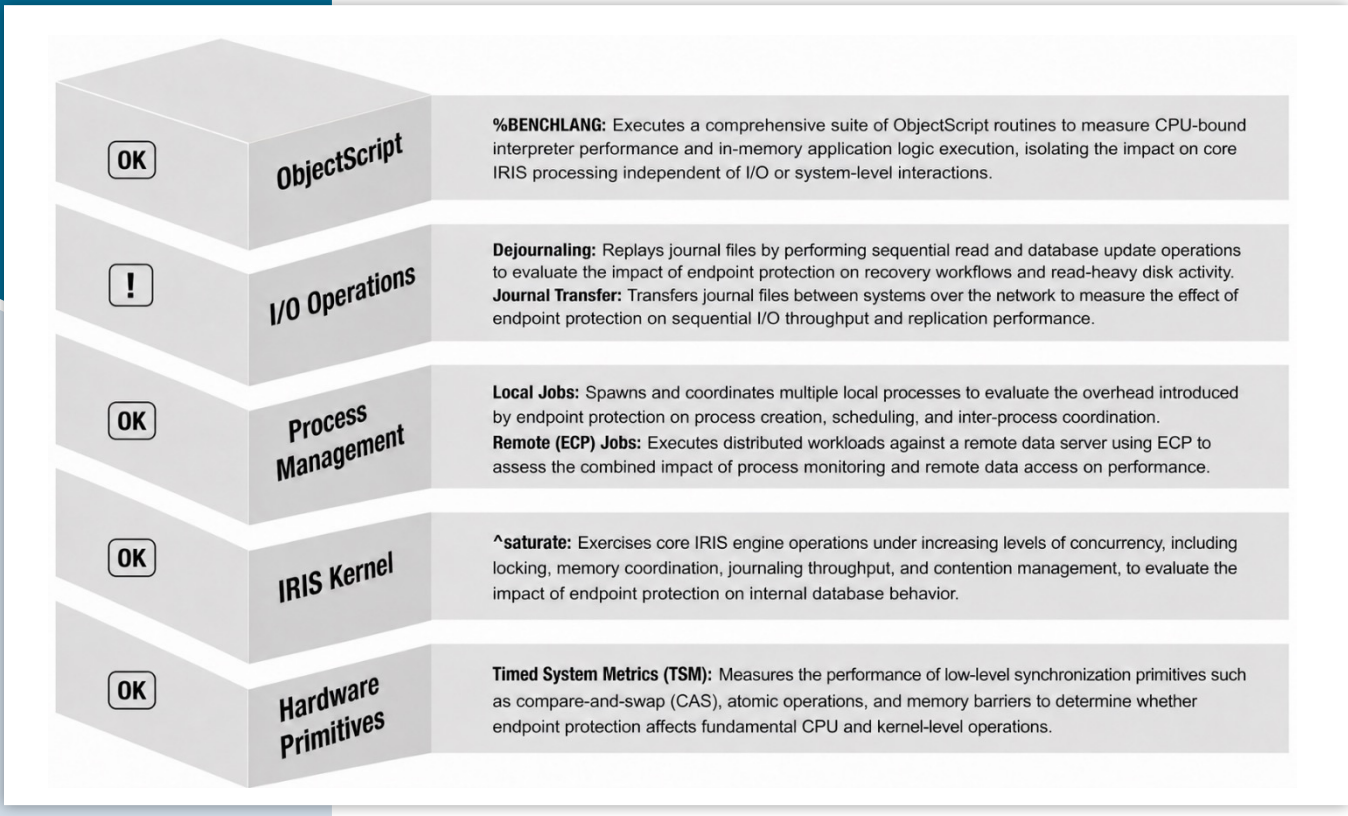


Figure 10: Test Results By Layer

Most observed differences fall within normal system variability. The use of multiple runs and averaged results helps mitigate this, but inherent noise in high-performance systems cannot be completely eliminated.

The presence of a few outliers, such as the Ubuntu XDR journal transfer result, highlights the need for environment-specific validation. These cases are not indicative of widespread issues but should be investigated to ensure optimal performance.

Recommendations

To optimize performance in environments where endpoint protection is deployed, apply the following practical steps:

- **Validate in a production-like environment:** Conduct benchmarking in staging environments that mirror production (data size, concurrency, journaling rates). Focus specifically on journaling, high-concurrency workloads, and failover scenarios.
- **Tune journaling paths (high impact):** Consider excluding InterSystems IRIS journal directories from real-time file scanning where appropriate. This is the most effective optimization identified in this study, reducing latency on critical write paths while preserving behavioral detection capabilities.
- **Monitor key performance indicators (KPIs):** Track metrics such as journal latency, replication lag, failover time, and CPU utilization before and after endpoint protection deployment.

- **Baseline before deployment:** Capture performance baselines (e.g., %SYS metrics, I/O latency, global references where applicable) to enable accurate comparison post-deployment.
- **Review agent policies:** Ensure endpoint protection configurations avoid overly aggressive file inspection or process instrumentation on database critical paths.
- **Validate failover and recovery workflows:** Test mirror failover and recovery (RPO/RTO) to confirm no unintended impact from endpoint protection.
- **Continuously monitor and iterate:** After deployment, implement ongoing monitoring to detect drift or regressions and adjust policies as needed.

InterSystems customers can confidently deploy endpoint protection on InterSystems IRIS database systems, as performance impact is minimal across most workloads. Modern endpoint protection platforms generally do not require the extensive exclusions that were common with traditional antivirus solutions.

Because journaling is a core InterSystems IRIS function and showed sensitivity in some environments, targeted tuning may be beneficial. Excluding journal directories from real-time file scanning can reduce latency on critical I/O paths while preserving behavioral analysis, telemetry collection, and anomaly detection capabilities.

These findings should guide validation in staging environments that reflect production workloads, with particular attention to journaling, compute-intensive applications, and high-concurrency scenarios.

Finally, ongoing monitoring should be implemented to detect any unexpected changes in performance following endpoint protection deployment.

Conclusions

This study evaluated the performance impact of modern endpoint protection platforms on InterSystems IRIS database workloads across a range of representative scenarios.

The key takeaways from this study are summarized below:

- Modern endpoint protection platforms can be deployed on InterSystems IRIS with minimal overall performance impact.
- Core database operations, including CPU-bound workloads and concurrency handling, remain unaffected.
- Journaling workloads may experience modest, environment-specific variability and should be validated.
- Operating system and agent implementation can influence performance outcomes more than endpoint protection presence alone.
- Proper tuning (e.g., journaling path exclusions) and validation ensure optimal performance without sacrificing security.

Our results show that endpoint protection can be deployed on InterSystems IRIS systems with minimal overall performance impact. CPU-bound operations and core database behaviors remain unchanged, with differences falling within expected system variability.

Journaling-related workloads showed modest, environment-specific variability, with one configuration (Ubuntu with Cortex XDR) exhibiting a larger impact (~8.3%) in journal transfer performance. This highlights the importance of validation in the target environment.

Overall, endpoint protection does not materially affect core execution or concurrency, while impacts on I/O-intensive workloads depend on system configuration and integration details.

These findings support deploying endpoint protection as part of a layered security strategy. Modern platforms provide behavioral and anomaly-based detection for advanced threats such as credential misuse and lateral movement and can be integrated with centralized logging and Security Information and Event Management (SIEM) systems for improved visibility and response.

Validation remains important, particularly for journaling-intensive workloads. Where needed, tuning inspection policies or reducing unnecessary file-based scanning can further optimize performance while preserving security capabilities.

In summary, endpoint protection should be considered a standard component of InterSystems IRIS deployments, with appropriate validation and tuning for the target environment. **The results presented here show that endpoint protection as a security strategy has no significant effect on the performance of InterSystems IRIS, validating its status as a modern, high-performance database able to process the most demanding workloads without compromise to security.**

Why InterSystems

InterSystems has been a trusted partner to health systems, health information exchanges, and government health programs for decades. Our expertise in health data interoperability, identity management, and clinical workflow integration gives us a deep understanding of what it takes to connect diverse organizations and foster collaboration. Leading health information exchanges are leveraging InterSystems technology to improve continuity of care and patient outcomes.

Learn more about InterSystems at **InterSystems.com**.

InterSystems®
Creative data technology

Products and services are subject to InterSystems terms and conditions. Information is provided for general informational purposes, may change without notice, and is provided without warranty. Product availability, functionality, and regulatory status may vary by offering, jurisdiction, and intended use.

For information regarding specific offerings, including availability and regulatory status, please contact your InterSystems Representative.

*For information regarding InterSystems approach to AI, visit **InterSystems AI Guidelines**.*

© 2026 InterSystems Corporation. All rights reserved. For trademark information, see **InterSystems Notices, Terms & Conditions**. 092426