BigFix Non-Functional Requirements:
A Checklist Approach

an HCL Product

HCL SOFTWARE

Document version 10.x.4

CONTENTS

List of Figures

REVISION HISTORY

Date	Version	Comments
November 8th, 2021	10.x.1	First draft.
November 30th, 2021	10.x.2	First publication with review comments incorporated.
December 17th, 2021	10.x.3	Minor content improvements.
July 10th, 2023	10.x.4	Root server controls.

Figure 1: Revision History

1 Introduction

Non-functional Requirements (NFRs) define the attributes of a system, versus a strict definition of the capability of the system. Two classic NFRs are performance (e.g., system utilization and throughput) and security (e.g., system protections). This document will provide a set of NFR checklists for BigFix.

We will first provide an architecture overview for the overall context of a BigFix deployment. We will then introduce the methodology applied, including classification, system components for consideration, and NFR classes. The overall checklists and associated detail will then be provided.

Some notes on the prescribed approach follow.

- In terms of performance, the intent of most tuning is to maximize throughput within the parameters of the system resources allocated.
- The agent is a special case of performance management, as it is typically throttled to drive a balance between maximum throughput and desired system impact. As a result, agent performance tuning is primarily dictated by the customer and their resource utilization goals.
- Monitoring solutions to determine impact may need to run over long time intervals to determine impact of scheduled events (e.g., “patch Tuesday”).
- In terms of security, there is a wide range of ethical hacking approaches that may be driven on top of BigFix (i.e., targeting the ecosystems BigFix is running in). While some approaches will be provided, they are not comprehensive (e.g., audit management for operating system administrative actions is not covered).
- The checklist is not intended to be a reference guide for OS kernel management, database administration, or hypervisor management. There are entire books devoted to these subjects. For VMware specifically, the References section provides performance best practices across specific VMware vSphere versions.
- The checklist is meant to complement the product reference documentation and is not intended to replace it. The product documentation is still the master reference for procedures and capability.

For the verification methods, the following resources will be referenced liberally. The associated URLS are provided in the References section, where applicable.

- The BigFix 10 Knowledge Center
  The product provided reference documentation.
- The BigFix 10 Capacity Planning Guide
  The product provided reference for BigFix capacity planning and tuning.
- The BigFix 10 Maintenance Guide
  The product provided reference for system and database maintenance.
- The BigFix Performance Toolkit
  A set of tools for performance management.
- The BigFix 10 Common Criteria Certification Guide
  A set of reference standards for the security and deployment of BigFix.
- The BigFix Architecture Guide
  A reference architecture document for BigFix, with specific details on security management. It is available only under a Non-Disclosure Agreement (NDA).

Note: This document is considered a work in progress. Recommendations will be refined and updated as new BigFix releases are available. While the paper in general is considered suitable for all BigFix Version 9.5.x releases, with new content as noted, it is best oriented towards BigFix Version 10.0.x onwards.

2 Architecture Overview

The following diagram provided a basic view of the BigFix architecture.

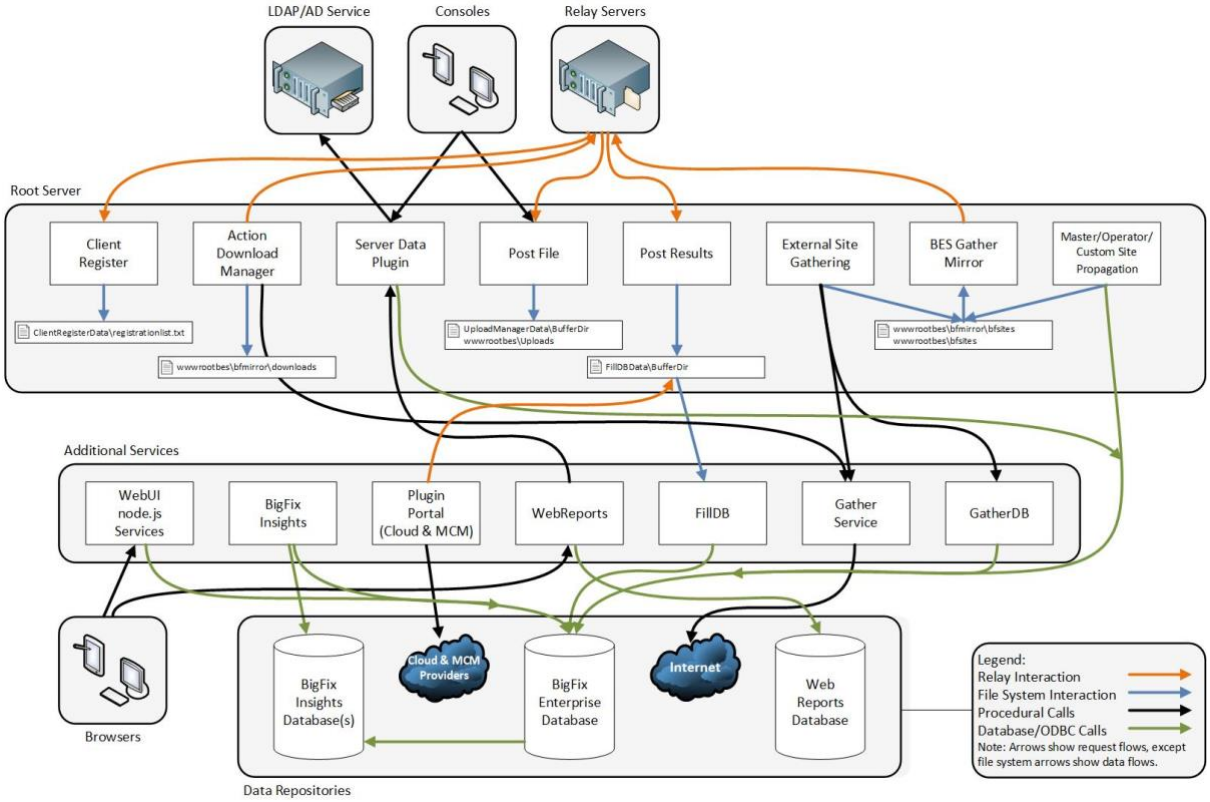

Figure 2: BigFix Architecture

The following components will be specifically referenced in this guide: Root Server, Console, the WebUI, Web Reports, Relays, Plugin Portal, and DBMS (in this case, MS SQL). The following components are considered out of scope but may be covered in a future revision: MCM, Insights, DSA, and Agents (the latter, as mentioned in the introduction).

For a more complete description and breakdown of the architectural components, the BigFix Knowledge Center and Capacity Planning Guide may be consulted.

3 Methodology

The methodology will break each checklist entity into five (5) parts.

1. The identifier.
   A unique ID that breaks down into a classification structure (to be described).
2. The title.
   A short text description of the entity.
3. The recommended application. This may be one of the following.
   - Mandatory.
     Essential for any deployment. Failure to adopt will result in issues.
   - Recommended.
     A best practice for any deployment. There may be valid reasons to defer or not apply.
   - Optional.
     Not essential, and not a best practice. May apply in some environments.
4. The detail for the specific entity (e.g., a basic description).
5. The verification mechanism for the specific entity. It answers the question: how do you know if it is enabled or operating well?

The classification structure for each unique identifier breaks down as follows.

1. The NFR type. This is one of the following.
   - PRF (for performance).
   - SEC (for security).
2. The associated component. This is one of the following.
   a. INF (for infrastructure, for example the operating system, storage, CPU, network, memory, virtualization hypervisor, etc.).
   b. BIG (for BigFix, for example the Root Server, Relay, Plugin Portal, etc.).
3. A unique two-digit numeric identifier.

The following figure shows a sample template for an individual entity.

ID: PRF-INF-XX	TITLE	MANDATORY
Detail:		
Verification:		

Figure 3: Sample Entity Template

4 Checklist Summary

The following tables provides the summary set of NFR checklist items for performance and security.

4.1 NFR Performance Checklist

	ID: Reference	Date Pre-Production	Date: Production
☐	PRF-INF-01: Storage Queue Depth		
☐	PRF-INF-02: Storage Latency		
☐	PRF-INF-03: Storage Operations/Second		
☐	PRF-INF-04: CPU Utilization		
☐	PRF-INF-05: CPU Latency		
☐	PRF-INF-06: CPU Privileged Time		
☐	PRF-INF-07: Network Health		
☐	PRF-INF-08: Network Topology		
☐	PRF-INF-09: Memory Health		
☐	PRF-INF-10: Hypervisor Latency		
☐	PRF-INF-11: Hypervisor vCPU Allocation		
☐	PRF-INF-12: Hypervisor vCPU Health		
☐	PRF-INF-13: Hypervisor Snapshots		
☐	PRF-INF-14: Linux IO Scheduler		
☐	PRF-INF-15: Linux Ulimit		
☐	PRF-INF-16: Linux Swappiness		
☐	PRF-INF-17: Windows Port Management		
☐	PRF-INF-18: DBMS Anti-Collocation		
☐	PRF-INF-19: DBMS Maximum Degree of Parallelism		
☐	PRF-INF-20: DBMS Cost Threshold for Parallelism		
☐	PRF-INF-21: DBMS Index Maintenance		
☐	PRF-INF-22: DBMS Plan Management		

☐	PRF-INF-23: DBMS Data Archiving		
☐	PRF-INF-24: DBMS Data Cardinality: Computers		
☐	PRF-INF-25: DBMS Data Cardinality: Open Actions		
☐	PRF-BIG-26: BigFix Capacity		
☐	PRF-BIG-27: Console Refresh		
☐	PRF-BIG-28: Console Cache Policy		
☐	PRF-BIG-29: WebUI Auto Update		
☐	PRF-BIG-30: FillDB Health Check		
☐	PRF-BIG-31: FillDB Parallelism		
☐	PRF-BIG-32: FillDB Buffer Directory		
☐	PRF-BIG-33: Relay Connections		
☐	PRF-BIG-34: Relay Scale TLRs		
☐	PRF-BIG-35: Relay Scale Leaf Nodes		
☐	PRF-BIG-36: Relay Health Check Dashboard		
☐	PRF-BIG-37: Plugin Portal Scale		
☐	PRF-BIG-38: Plugin Portal Parallelism		
☐	PRF-BIG-39: MDM Docker Configuration		
☐	PRF-BIG-40: Agent Heartbeat		
☐	PRF-BIG-41: Agent Minimum Report Interval		
☐	PRF-BIG-42: Target by List Limits		

Figure 4: NFR Performance Checklist

4.2 NFR Security Checklist

	ID: Reference	Date Pre-Production	Date: Production
☐	SEC-INF-01: OS Currency		
☐	SEC-INF-02: DBMS Currency		
☐	SEC-INF-03: DBMS Force Encryption		
☐	SEC-INF-04: DBMS Common Criteria		
☐	SEC-INF-05: DBMS Service Protection		
☐	SEC-INF-06: ODBC Strong Encryption		
☐	SEC-INF-07: Nmap Port Scan		
☐	SEC-INF-08: Nmap Security Scan		
☐	SEC-BIG-09: BigFix Currency		
☐	SEC-BIG-10: FIPS 140-2		
☐	SEC-BIG-11: MLE 2048		
☐	SEC-BIG-12: Enhanced Security		
☐	SEC-BIG-13: LDAP/SAML Integration		
☐	SEC-BIG-14: Masthead		
☐	SEC-BIG-15: Authenticating Relays		
☐	SEC-BIG-16: Relay Diagnostics Page		
☐	SEC-BIG-17: Password Protection		
☐	SEC-BIG-18: Login Protection		
☐	SEC-BIG-19: Action Confirmation		
☐	SEC-BIG-20: Agent Secure Registration		
☐	SEC-BIG-21: Admin Key Protection		

Figure 5: NFR Security Checklist

5 Checklist Detail: Performance

The performance detail items will be provided.

5.1 PRF-INF-01: Storage Queue Depth

ID: PRF-INF-01	Storage Queue Depth	Mandatory
Detail: Storage should exhibit queues no greater than a queue depth of one (1) per physical device.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Physical Disk → Current Disk Queue Length • Value(s): ≤ 1 per physical device.		

Figure 6: PRF-INF-01: Storage Queue Depth

5.2 PRF-INF-02: Storage Latency

ID: PRF-INF-02	Storage Latency	Mandatory
Detail: Storage latency should be at or below one (1) millisecond under load.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Physical Disk → Avg. Disk sec/Transfer • Value(s): ≤ 1ms per physical device.		

Figure 7: PRF-INF-02: Storage Latency

5.3 PRF-INF-03: Storage Operations/Second

ID: PRF-INF-03	Storage Operations/Second	Mandatory
Detail: Storage should support at least five thousand (5,000) operations per second. Note if the prior queue depth and latency requirements are met, the operations/second should be acceptable.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Physical Disk → Disk Transfers/sec • Value(s): No saturation should be in evidence under load (e.g., flatlining). For the DBMS storage devices, support for 5,000 disk transfers per second should be achievable. The Capacity Planning Guide provides a set of IO workload profiles for standalone IO benchmarks. These workload profiles may be deployed by tools like Iometer.		

Figure 8: PRF-INF-03: Storage Operations/Second

5.4 PRF-INF-04: CPU Utilization

ID: PRF-INF-04	CPU Utilization	Mandatory
Detail: CPU utilization for a hyperthreaded environment should be in the range of 50 to 80%. Values below or above this may indicate a poor allocation (either too few resources, or too many). Note for virtual deployments, too many resources can lead to hypervisor scheduling overhead.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Processor → % Processor Time • Value(s): o For virtual deployments: ≥ 50% and ≤ 80% aggregate. o For physical deployments: ≤ 80% aggregate.		

Figure 9: PRF-INF-04: CPU Utilization

5.5 PRF-INF-05: CPU Latency

ID: PRF-INF-05	CPU Latency	Mandatory
Detail: CPU queue depth should never exceed the low single digits. High values indicate a resource or workload problem (e.g., high kernel utilization).		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): System → Processor Queue Length • Value(s): ≤ 4		

Figure 10: PRF-INF-05: CPU Latency

5.6 PRF-INF-06: CPU Privileged Time

ID: PRF-INF-06	CPU Privileged Time	Mandatory
Detail: CPU privileged time should be below 10% of the aggregate CPU utilization. If it is not, it can indicate kernel overhead due to stress or misconfiguration.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Processor → % Privileged Time • Value(s): ≤ 10% of aggregate CPU.		

Figure 11: PRF-INF-06: CPU Privileged Time

5.7 PRF-INF-07: Network Health

ID: PRF-INF-07	Network Health	Mandatory
Detail: Network health should be validated for key interfaces (saturation, dropped packets, etc.).		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): o Network Interface → Packets Outbound Discarded o Network Interface → Packets Outbound Errors o Network Interface → Packets Received Discarded o Network Interface → Packets Received Errors • Value(s): All counters should approach zero per interface.		

Figure 12: PRF-INF-07: Network Health

5.8 PRF-INF-08: Network Topology

ID: PRF-INF-08	Network Topology	Mandatory
Detail: Routing and topology should be optimal (e.g., one “hop” between key components, such as Root Server to DBMS, Root Server to Storage Appliance, Console to Root Server, etc.).		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): o Basic ping (*NIX): ping –c 10 <ip> Basic ping (Windows) ping -n 10 <ip> o Flood ping (*NIX): ping –f –c 100000 –s 1500 –l 4 <ip> Flood ping (Windows): ping -w 10 -n 100000 -l 1500 <ip> o Trace route: tracert <ip> • Value(s): Single hop between server components with ping and route times ≤ 2ms.		

Figure 13: PRF-INF-08: Network Topology

5.9 PRF-INF-09: Memory Health

ID: PRF-INF-09	Memory Health	Mandatory
Detail: Memory is there to be used. High memory utilization is typically not a problem unless paging is in evidence. Ensure there is no paging, and possibly shift the memory allocation (e.g., pinning DBMS memory) to compensate.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): Paging File → % Usage Peak • Value(s): ≤ 5% of aggregate physical memory.		

Figure 14: PRF-INF-09: Memory Health

5.10 PRF-INF-10: Hypervisor Latency

ID: PRF-INF-10	Hypervisor Latency	Mandatory
Detail: The hypervisor (if applicable) latency sensitivity setting should be enabled for the BigFix guests.		
Verification: • Reference(s): VMware references. Other hypervisor references may apply. • Performance Setting(s): VM Options → Latency Sensitivty • Value(s): Set to “High”.		

Figure 15: PRF-INF-10: Hypervisor Latency

5.11 PRF-INF-11: Hypervisor vCPU Allocation

ID: PRF-INF-11	Hypervisor Latency	Mandatory
Detail: The hypervisor (if applicable) virtual CPU allocation should align with peak workloads (e.g., an allocation with 2 vCPU headroom over peak). Note: For VMware it is recommended to disable CPU Hot-Add. This disables automatic NUMA sizing that may introduce performance problems.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit, VMware references. Other hypervisor references may apply. • Performance Setting(s): VM Options → Virtual CPU Allocation • Value(s): Configure based on the endpoint deployment. Note the BigFix Performance Toolkit provides a capacity planning tool to simplify calculations.		

Figure 16: PRF-INF-11: Hypervisor vCPU Allocation

5.12 PRF-INF-12: Hypervisor vCPU Health

ID: PRF-INF-12	Hypervisor Latency	Mandatory
Detail: The hypervisor (if applicable) virtual CPU ready time should be on average 0-50ms. Anything over 1000ms will exhibit performance issues.		
Verification: • Reference(s): BigFix Capacity Planning Guide, VMware references. Other hypervisor references may apply. • Performance Counter(s): VM Monitor → CPU Ready • Value(s): Average CPU ready in the interval [0ms,50ms].		

Figure 17: PRF-INF-12: Hypervisor vCPU Health

5.13 PRF-INF-13: Hypervisor Snapshots

ID: PRF-INF-13	Hypervisor Snapshots	Mandatory
Detail: The hypervisor (if applicable) should employ minimal snapshots. Snapshots generate block storage chains that can seriously degrade performance. A best practice can be snapshot chains of one (1) for upgrade operations, etc.		
Verification: • Reference(s): BigFix Capacity Planning Guide, VMware references. Other hypervisor references may apply. • Performance Counter(s): VM Monitor → Snapshots • Value(s): Snapshot chain ≤ 1 for tree depth.		

Figure 18: PRF-INF-13: Hypervisor Snapshots

5.14 PRF-INF-14: Linux IO Scheduler

ID: PRF-INF-14	Linux IO Scheduler	Mandatory
Detail: The Linux IO scheduler should be configured for noop or deadline if running in a virtual environment.		
Verification: • Reference(s): BigFix Capacity Planning Guide • Performance Setting(s): /sys/block/<device>/queue/scheduler • Value(s): Either the “noop” or “deadline” policy should be configured.		

Figure 19: PRF-INF-14: Linux IO Scheduler

5.15 PRF-INF-15: Linux Ulimit

ID: PRF-INF-15	Linux Ulimit	Mandatory
Detail: The Linux “ulimit” parameter should be configured for all BigFix servers (root, DBMS, relays, plugin portal, etc.).		
Verification: • Reference(s): BigFix Capacity Planning Guide • Performance Setting(s): ulimit -a • Value(s): The value should be unlimited or 65536.		

Figure 20: PRF-INF-15: Linux Ulimit

5.16 PRF-INF-16: Linux Swappiness

ID: PRF-INF-16	Linux Swappiness	Recommended
Detail: The Linux “swappiness” configuration parameter should be set for any Linux based DBMS server.		

Verification:
• Reference(s): BigFix Capacity Planning Guide
• Performance Setting(s): Kernel parameter → vm.swappiness
• Value(s):
  ○ Value for dedicated DBMS server: 0
  ○ Value for a DBMS server collocated with the root server: 10		

Figure 21: PRF-INF-16: Linux Swappiness

5.17 PRF-INF-17: Windows Port Management

ID: PRF-INF-17	Windows Port Management	Mandatory
Detail: System network throughput may be optimized by adjusting the Windows “time wait” interval for recycling TCP/IP resources.		
Verification: • Reference(s): BigFix Capacity Planning Guide, Microsoft Technote: URL • Performance Setting(s): System Registry → TcpTimedWaitDelay • Value(s): 30 seconds.		

Figure 22: PRF-INF-17: Windows Port Management

5.18 PRF-INF-18: DBMS Anti-Collocation

ID: PRF-INF-18	DBMS Anti-Collocation	Recommended
Detail: In general, BigFix can support a wide range of reference architectures. Two reasons support DBMS anti-collocation (i.e., the DBMS on a dedicated instance independent from the Root Server). 1. Database Administrator (DBA) support. 2. Virtualization and the ability to run smaller, dedicated VMs.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Maintenance Guide • Performance Setting(s): Not applicable. • Value(s): Collocate or anti-collocate based on installation specifics.		

Figure 23: PRF-INF-18: DBMS Anti-Collocation

5.19 PRF-INF-19: DBMS Maximum Degree of Parallelism

ID: PRF-INF-19	DBMS Maximum Degree of Parallelism	Mandatory
Detail: The DBMS Maximum Degree of Parallelism (MAXDOP) controls parallelism to maximize throughput. The setting is based on the processor topology and aligns with Microsoft reference values. Benchmarks utilizing these settings have demonstrated increased throughput with reduced system load.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Setting(s): SQL Server Management Studio → EXEC sp_configure 'show advanced options', 1; (note: this will allow you to inspect the current settings) • Value(s): Per the BigFix Knowledge Center reference. Note: BigFix provides tooling to manage the configuration for installation and upgrade. In the event a standalone DBMS server has been built, it should be ensured the values are in effect.		

Figure 24: PRF-INF-19: DBMS Maximum Degree of Parallelism

5.20 PRF-INF-20: DBMS Cost Threshold for Parallelism

ID: PRF-INF-20	DBMS Cost Threshold for Parallelism	Mandatory
Detail: The DBMS Cost Threshold for Parallelism (CTFP) controls execution plan parallelism based on the optimizer cost. A value of 50 should be set.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Setting(s): SQL Server Management Studio → EXEC sp_configure 'show advanced options', 1; (note: this will allow you to inspect the current settings) • Value(s): 50. Note: BigFix provides tooling to manage the configuration for installation and upgrade. In the event a standalone DBMS server has been built, it should be ensured the values are in effect.		

Figure 25: PRF-INF-20: DBMS Cost Threshold for Parallelism

5.21 PRF-INF-21: DBMS Index Maintenance

ID: PRF-INF-21	DBMS Index Maintenance	Mandatory
Detail: BigFix installs an index maintenance script that reorganize and rebuilds indexes based on defined fragmentation thresholds. The script needs to be installed, enabled, and healthy.		

Verification:
• Reference(s): BigFix Maintenance Guide, Blog: BigFix 10 Infrastructure Monitoring
• Performance Setting(s): SQL Server Management Studio → SQL Server Agent → "BFEnterprise Full Database Index Reorganization" Job
• Value(s):
  ○ Ensure the BigFix provided job is installed.
  ○ Ensure the job is running at least nightly.
  ○ Ensure the job is running successfully (i.e., without errors or locking issues). The log generated by the job will show issues, along with a non-successful return code.
  ○ Optional: Verify the index fragmentation after the job has run. For large indexes (those > 1000 pages) the fragmentation should be ≤ 5%. The referenced blog provides a query to obtain the fragmentation values.		

Figure 26: PRF-INF-21: DBMS Index Maintenance

5.22 PRF-INF-22: DBMS Plan Management

ID: PRF-INF-22	DBMS Index Maintenance	Recommended
Detail: Database management queries and plans should be inspected to ensure workloads are healthy. A general rule is any query over one (1) second should be understood in terms of source and impact.		
Verification: • Reference(s): BigFix Maintenance Guide, Blog: BigFix 10 Infrastructure Monitoring • Performance Counter(s): Generate the MS SQL package cache results per the blog. • Value(s): o BigFix queries over one (1) second should be understood in terms of source and impact. o Custom queries over one (1) second should be understood, including lock impact. Ideally, these queries should be refactored.		

Figure 27: PRF-INF-22: DBMS Index Maintenance

5.23 PRF-INF-23: DBMS Data Archiving

ID: PRF-INF-23	DBMS Data Archiving	Mandatory
Detail: Data archiving ensures operational health of the BigFix deployment.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL URL • Performance Setting(s): o Computer Removal Utility o Audit Cleanup Utility • Value(s): Schedule the utilities to run at least weekly per business standards.		

Figure 28: PRF-INF-23: DBMS Data Archiving

5.24 PRF-INF-24: DBMS Data Cardinality: Computers

ID: PRF-INF-24	DBMS Data Cardinality: Computers	Mandatory
Detail: BigFix can support up to 300,000 computer objects for a single Root Server deployment.		
Verification: • Reference(s): BigFix Capacity Planning Guide • Performance Counter(s): SQL Server Management Studio → “select count(*) from dbo.computers where isDeleted=0“ • Value(s): ≤ 300,000		

Figure 29: PRF-INF-24: DBMS Data Cardinality: Computers

5.25 PRF-INF-25: DBMS Data Cardinality: Open Actions

ID: PRF-INF-25	DBMS Data Cardinality: Open Actions	Mandatory
Detail: BigFix can support up to 5,000 open actions for a single Root Server deployment.		
Verification: • Reference(s): BigFix Capacity Planning Guide • Performance Counter(s): Relevance → number of bes actions whose("Open" = state of it) • Value(s): ≤ 5,000		

Figure 30: PRF-INF-25: DBMS Data Cardinality: Open Actions

5.26 PRF-BIG-26: BigFix Capacity

ID: PRF-BIG-26	BigFix Capacity	Mandatory
Detail: BigFix provides a capacity planning reference and associated tools. The tools should be used along with system monitoring to ensure the deployment is within specification.		

Verification:
• Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit
• Performance Counter(s): Performance Toolkit → MXCapacity
• Value(s):
  o Select suitable parameters for the MXCapacity tool.
  o Compare the recommended values with the deployment. Adjust as appropriate.		

Figure 31: PRF-BIG-26: BigFix Capacity

5.27 PRF-BIG-27: Console Refresh

ID: PRF-BIG-27	Console Refresh	Mandatory
Detail: The BigFix Console Fixlet Refresh interval defaults to fifteen (15) seconds and may be too aggressive for large installations.		

Verification:
• Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL
• Performance Counter(s): Console Preferences → Refresh
• Value(s):
  ○ One second per 1,000 endpoints, with a minimum value of 15s.
  ○ For example, a 300,000 end point deployment should have a value ≥ 300s.		

Figure 32: PRF-BIG-27: Console Refresh

5.28 PRF-BIG-28: Console Cache Policy

ID: PRF-BIG-28	Console Cache Policy	Mandatory
Detail: The BigFix Console Cache Policy should be customized for the deployment.		

Verification:
• Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL
• Performance Counter(s): Console Preferences → Caching
• Value(s):
  ○ Dependent on the console instance.
  ○ The goal should be a full cache with a moderate expiration policy.		

Figure 33: PRF-BIG-28: Console Cache Policy

5.29 PRF-BIG-29: WebUI Auto Update

ID: PRF-BIG-29	WebUI Auto Update	Recommended
Detail: The BigFix WebUI may automatically update. This may lead to unplanned updates and production impact. The feature should be disabled in favor of controlled updates.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Performance Counter(s): Update Manager → Auto Update • Value(s): Set to “Off”.		

Figure 34: PRF-BIG-29: WebUI Auto Update

5.30 PRF-BIG-30: FillDB Health Check

ID: PRF-BIG-30	FillDB Health Check	Mandatory
Detail: The FillDB daemon is critical to the health of BigFix. A performance analyzer with a built-in health check is available. The performance analyzer should be run, and the results verified to be within specification.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit • Performance Counter(s): MXFillDBPerf --input <FillDB performance log> –healthcheck –stats • Value(s): The following values should have a health check of “Good” or “Great”. o Parallel DB Update (Short Batch) o Parallel DB Update o Batch Rate o Parallel Parsing		

Figure 35: PRF-BIG-30: FillDB Health Check

5.31 PRF-BIG-31: FillDB Parallelism

ID: PRF-BIG-31	FillDB Parallelism	Optional
Detail: For healthy environments with additional capacity, it is possible to “turbo charge” FillDB. The parallelism may be increased to drive higher throughput rates and responsiveness. This should be done with careful monitoring as increased parallelism can lead to throughput degradation.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Performance Toolkit, Blog: BigFix FillDB Performance • Performance Settings(s): o FillDB Configuration → NumberOfParsingThreads o FillDB Configuration → NumberOfDBUpdatingThreads o FillDB Configuration → NumberOfParsingThreadsForQuery o FillDB Configuration → NumberOfDBUpdatingThreadsForQuery • Value(s): o Values should be in the interval range [3,5] depending on core allocation and available IOPS with 1ms latency for the database storage device(s). o For example, with 100k IOPS @ <1ms latency, a setting of five parsing threads and 5 update threads (across all four settings) works well. o Note, increased parallelism with insufficient cores or IOPS can lead to degradation.		

Figure 36: PRF-BIG-31: FillDB Parallelism

5.32 PRF-BIG-32: FillDB Buffer Directory

ID: PRF-BIG-32	FillDB Buffer Directory	Optional
Detail: The FillDB Buffer Directory (aka the “BufferDir”) configuration should be verified.		

Verification:
• Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL
• Performance Settings(s):
  ○ FillDB Configuration → BufferDirectoryMaxSize
  ○ FillDB Configuration → BufferDirectoryMaxCount
• Value(s):
  ○ BufferDirectoryMaxSize: The default value is 3MB.
BufferDirectoryMaxCount: The default value is 10,000.
 	
These values should be changed only if there is evidence of report data loss or if it is desired to run benchmarks or saturation tests on the deployment (e.g., in a pre-production performance verification environment).		

Figure 37: PRF-BIG-32: FillDB Buffer Directory

5.33 PRF-BIG-33: Relay Connections

ID: PRF-BIG-33	Relay Connections	Optional
Detail: The BESRelay_HTTPServer_MaxConnections configuration parameter will determine the maximum number of concurrent connections for the relay. This defaults to 2048 on Windows and 512 on Linux. It may be changed for extreme high concurrency scenarios.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Settings(s): Relay Configuration → BESRelay_HTTPServer_MaxConnections • Performance Counter(s): netstat -a • Value(s): Monitoring is recommended to determine the connection levels. If the threshold is being reached, incrementally increase by 10%. Note the BigFix 10.0.5 release introduces efficiencies for relay connection management and the TLS handshake. BigFix 11 enables TLS 1.3 which provides further handshake efficiencies.		

Figure 38: PRF-BIG-33: Relay Connections

5.34 PRF-BIG-34: Relay Scale TLRs

ID: PRF-BIG-34	Relay Scale TLRs	Mandatory
Detail: Top level relays are generally recommended once you approach 40,000 endpoints or over 100 relays (whichever comes first). A top-level relay should manage no more than 40,000 endpoints or 120 relays (whichever comes first).		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Relay Health Dashboard • Value(s): Ensure all is well via the Relay Health Dashboard.		

Figure 39: PRF-BIG-34: Relay Scale TLRs

5.35 PRF-BIG-35: Relay Scale Leaf Nodes

ID: PRF-BIG-35	Relay Scale Leaf Nodes	Mandatory
Detail: High scale leaf node relays can manage 5,000 agents/endpoints (a 1:5,000 ratio). The management ratio should be verified.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Relay Health Dashboard • Value(s): Ensure all is well via the Relay Health Dashboard.		

Figure 40: PRF-BIG-35: Relay Scale Leaf Nodes

5.36 PRF-BIG-36: Relay Health Check Dashboard

ID: PRF-BIG-36	Relay Health Check Dashboard	Mandatory
Detail: The Relay health check dashboard should be monitored for continuous health.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Relay Health Dashboard • Value(s): Ensure all is well via the Relay Health Dashboard.		

Figure 41: PRF-BIG-36: Relay Health Check Dashboard

5.37 PRF-BIG-37: Plugin Portal Scale

ID: PRF-BIG-37	Plugin Portal Scale	Mandatory
Detail: The number of objects managed via a single Plugin Portal instance should not exceed 75,000. It is possible to deploy more than one Plugin Portal instance. See the product references below for more detail on multi-Portal deployments.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Plugin Portal → Aggregate count of all managed objects • Value(s): ≤ 75,000		

Figure 42: PRF-BIG-37: Plugin Portal Scale

5.38 PRF-BIG-38: Plugin Portal Parallelism

ID: PRF-BIG-38	Plugin Portal Parallelism	Optional
Detail: The Plugin Portal has a configurable thread limit. For example: • Windows: HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\BigFix\EnterpriseClient\Settings\Client_BESPluginPortal_Performance_ThreadLimit • Linux: BESPluginPortal_Performance_ThreadLimit The default threading value for relevance evaluation is automatically set to the number of cores available. The maximum allowed value is 128. Given the configuration is automatically managed for new deployments in BigFix 10.0.4, it is recommended legacy deployments remove this setting so it may be automatically managed.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Plugin Portal → BESPluginPortal_Performance_ThreadLimit • Value(s): Automatically managed based on available Plugin Portal resources.		

Figure 43: PRF-BIG-38: Plugin Portal Parallelism

5.39 PRF-BIG-39: MDM Docker Configuration

ID: PRF-BIG-39	MDM Docker Configuration	Mandatory
Detail: The MDM Docker instance requires some specific configuration changes to manage scale.		

Verification:
• Reference(s): BigFix Capacity Planning Guide
• Performance Counter(s): Counters and their associated values are provided below.
• Value(s): Set the following configuration values for the docker instance.
  • /etc/security/limits.conf:
  root soft nofile 100000
  root hard nofile 100000
  root soft nproc 100000
  • /etc/sysctl.conf:
  net.core.rmem_default=1000000
  net.core.wmem_default=1000000
  net.core.rmem_max=1000000
  net.core.wmem_max=1000000
  net.ipv4.tcp_rmem=4096 87380 167177216
  net.ipv4.tcp_wmem=4096 65536 167177216		

Figure 44: PRF-BIG-39: MDM Docker Configuration

5.40 PRF-BIG-40: Agent Heartbeat

ID: PRF-BIG-40	Agent Heartbeat	Mandatory
Detail: The agent sends a heartbeat to the server (essentially, reporting in), every 15 minutes. As a BigFix deployment scales, the heartbeat activity can become significant. For example, if 250,000 agents are reporting every 15 minutes, that is over 278 heartbeats per second! In order to mitigate this, the general recommendation is to set the heartbeat interval to 15 minutes for every 10,000 agents.		
Verification: • Reference(s): BigFix Capacity Planning Guide • Performance Counter(s): BigFix Console → Preferences • Value(s): The general recommendation is to set the heartbeat interval to 15 minutes for every 10,000 agents. For example, for a 250,000 agent deployment, this would mean a heartbeat on the order of 6 hours		

Figure 45: PRF-BIG-40: Agent Heartbeat

5.41 PRF-BIG-41: Agent Minimum Report Interval

ID: PRF-BIG-41	Agent Minimum Report Interval	Recommended
Detail: The agent setting _BESClient_Report_MinimumInterval should be set appropriately, especially for large installations.		
Verification: • Reference(s): BigFix Capacity Planning Guide, BigFix Knowledge Center: URL • Performance Counter(s): Agent → _BESClient_Report_MinimumInterval • Value(s): The default vale is 60s. Changes to the value are determined by business priorities. o To increase the frequency of reports, the value may be reduced to 30s. o To decrease the server pressure, the value may be increase to 300s.		

Figure 46: PRF-BIG-41: Agent Minimum Report Interval

5.42 PRF-BIG-42: Target by List Limits

ID: PRF-BIG-42	Target by List Limits	Recommended
Detail: This set of advanced options provides target limits and may be viewed as a form of throttle control.		

Verification:
• Reference(s): BigFix Knowledge Center: URL
• Performance Counter(s): Agent → _BESClient_Report_MinimumInterval
  ○ BES Admin Tool → targetBySpecificListLimit
  Specifies the maximum number of computers that can be targeted by individual selection.
  Default: 10,000
  ○ BES Admin Tool → targetBySpecificListWarning
  Specifies the threshold for the number of computers that can be targeted by individual selection before the console displays a warning message.
  Default: 1,000
  ○ BES Admin Tool → targetByListSizeLimit
  Specifies the maximum number of bytes that can be supplied when targeting by textual list of computer names.
  Default: 100,000
• Value(s): Values may be chosen based on business and operator controls. Some best practices follow.
  ○ The values should not be increased from the default.
  ○ The targetBySpecificListLimit may be reduced to provide further control, while still providing ease of administration. For example, a value of 10% of the expected target size can prove beneficial.
  ○ The targetBySpecificListWarning can then be set appropriately. For example, a value in the interval [10%,100%] of the targetBySpecificListLimit.		

Figure 47: PRF-BIG-42: Target by List Limits

6 Checklist Detail: Security

The security detail items will be provided.

6.1 SEC-INF-01: OS Currency

ID: SEC-INF-01	OS Currency	Recommended
Detail: A general security best practice is to be on the vendor's most recent operating system distribution with recent patches applied. For operating system instances with BigFix objects deployed on them, the supported operating system reference should be consulted. BigFix may be utilized for patch management.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Cross reference operating system levels with the support matrix.		

Figure 48: SEC-INF-01: OS Currency

6.2 SEC-INF-02: DBMS Currency

ID: SEC-INF-02	DBMS Currency	Recommended
Detail: A general security best practice is to be on the vendor's most recent software distribution with recent patches applied. For the DBMS, the latest supported version with patches applied should be installed. The supported DBMS reference should be consulted.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Cross reference DBMS levels with the support matrix.		

Figure 49: SEC-INF-02: DBMS Currency

6.3 SEC-INF-03: DBMS Force Encryption

ID: SEC-INF-03	DBMS Force Encryption	Optional
Detail: The force encryption option for MS SQL ensures encrypted communication to the DBMS engine. There is associated but manageable performance overhead. Enablement is via MS SQL best practices: URL.		
Verification: • Reference(s): BigFix Capacity Planning Guide (MS SQL Secure Settings) • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 50: SEC-INF-03: DBMS Force Encryption

6.4 SEC-INF-04: DBMS Common Criteria

ID: SEC-INF-04	DBMS Common Criteria	Optional
Detail: Common criteria compliance includes such dimensions as Residual Information Protection (RIP), login audit, and authorization changes. There is associated but manageable performance overhead. Enablement is via MS SQL best practices: URL.		
Verification: • Reference(s): BigFix Capacity Planning Guide (MS SQL Secure Settings) • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 51: SEC-INF-04: DBMS Common Criteria

6.5 SEC-INF-05: DBMS Service Protection

ID: SEC-INF-05	DBMS Service Protection	Optional
Detail: By default, the MS SQL instance is configured to support up to 32,767 connections. This has the potential to degrade the service to the point it may effectively yield a Denial of Service (DoS) attack. As a result, a practical limit of 1,000 is recommended. Enablement is via MS SQL best practices: URL.		
Verification: • Reference(s): BigFix Capacity Planning Guide (MS SQL Secure Settings) • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 52: SEC-INF-05: DBMS Service Protection

6.6 SEC-INF-06: ODBC Strong Encryption

ID: SEC-INF-06	ODBC Strong Encryption	Optional
Detail: This option will enforce strong encryption for the ODBC connection. It is recommended if the database server is anti-collocated with the BigFix server(s) (e.g., the Root Server, Web Reports, or WebUI). There is associated but manageable performance overhead. Enablement is via MS SQL best practices: URL.		
Verification: • Reference(s): BigFix Capacity Planning Guide (MS SQL Secure Settings) • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 53: SEC-INF-06: ODBC Strong Encryption

6.7 SEC-INF-07: Nmap Port Scan

ID: SEC-INF-07	Nmap Port Scan	Recommended
Detail: Run nmap against the BigFix servers and relay infrastructure, including the Plugin Portal. The following invocation will provide a port scan: “nmap -p0- -A -T4 <host>”. For example, it should be ensured the ftp and ssh services are disabled.		
Verification: • Reference(s): nmap -p0- -A -T4 <host> • Procedure: Verify only required services are available. For additional security, ensure firewalls and whitelists are in place. For example, only necessary users and services should be able to access the BigFix root server on port 52311.		

Figure 54: SEC-INF-07: Nmap Port Scan

6.8 SEC-INF-08: Nmap Security Scan

ID: SEC-INF-08	Nmap Security Scan	Recommended
Detail: Run nmap against the BigFix servers and relay infrastructure, including the Plugin Portal. The following invocation will provide a security scan of known certificates and ciphers: “nmap -p <port> --script ssl-cert,ssl-enum-ciphers <host>”.		
Verification: • Reference(s): nmap -p <port> --script ssl-cert,ssl-enum-ciphers <host> • Procedure: Verify only required certificates and ciphers are available.		

Figure 55: SEC-INF-08: Nmap Security Scan

6.9 SEC-BIG-09: BigFix Currency

ID: SEC-BIG-09	BigFix Currency	Mandatory
Detail: A general security best practice is to be on the most recent distribution of BigFix with recent patches applied.		
Verification: • Reference(s): BigFix Download Center: URL • Procedure: Cross reference the product distribution with the support matrix.		

Figure 56: SEC-BIG-09: BigFix Currency

6.10 SEC-BIG-10: FIPS 140-2

ID: SEC-BIG-10	BigFix FIPS 140-2	Optional
Detail: The Federal Information Processing Standard (FIPS) is a set of standards and certifications for system security. You may use the following product reference to enable FIPS 140-2 for the BigFix Root Server: URL. It may also be enabled for the Web Reports server.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Enablement with monitoring to ensure performance degradation is manageable. The possibility of performance degradation is why this is optional.		

Figure 57: SEC-BIG-10: FIPS 140-2

6.11 SEC-BIG-11: MLE 2048

ID: SEC-BIG-11	MLE 2048	Mandatory
Detail: Message Level Encryption (MLE) enables clients to encrypt upstream data using a combination of an RSA public/private key-pair and an AES session key. While MLE 4096 is possible, MLE 2048 is recommended as it is both secure and performs with manageable overhead.		
Verification: • Reference(s): Blog: BigFix MLE Enablement, BigFix Knowledge Center: URL • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 58: SEC-BIG-11: MLE 2048

6.12 SEC-BIG-12: Enhanced Security

ID: SEC-BIG-12	Enhanced Security	Mandatory
Detail: Prior to BigFix 11, enhanced security enforces SHA-256 for all digital signatures and TLS 1.2 for all HTTPS communications. For BigFix 11, it enforces SHA-384 and TLS 1.3.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Enablement with monitoring to ensure degradation is manageable.		

Figure 59: SEC-BIG-12: Enhanced Security

6.13 SEC-BIG-13: LDAP/SAML Integration

ID: SEC-BIG-13	LDAP/SAML Integration	Recommended
Detail: It is generally recommended to integrate with a Lightweight Directory Access Protocol (LDAP) server. BigFix v9.5.5 onwards supports SAML V2.0 authentication via LDAP-backed SAML identity providers for the Web Reports, Web UI, and Console components.		
Verification: • Reference(s): BigFix Knowledge Center: URL (LDAP), BigFix Knowledge Center: URL (SAML) • Procedure: Enablement based on enterprise directory standards.		

Figure 60: SEC-BIG-13: LDAP/SAML Integration

6.14 SEC-BIG-14: Masthead

ID: SEC-BIG-14	Masthead	Mandatory
Detail: Based on the upgrade policies of the BigFix servers, the following Masthead settings may not be enabled. These settings enforce HTTPS (TLS) based registration policies. • minimumsupportedclient • minimumsupportedrelay		

Verification:
• Reference(s): BigFix Knowledge Center: URL
• Procedure: Ensure the following values are set appropriately.
  ○ BigFix Administration Tool → minimumsuppportedclient = 9.0
  ○ BigFix Administration Tool → minimumsuppportedrelay = 9.5.6		

Figure 61: SEC-BIG-14: Masthead

6.15 SEC-BIG-15: Authenticating Relays

ID: SEC-BIG-15	Authenticating Relays	Recommended
Detail: Authenticating relays enforce HTTPS (TLS) communication through the relay chain. It is especially important for internet facing relays.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Enablement with monitoring to ensure degradation is manageable. Note: The BigFix 10.0.5 release includes major improvements for the TLS communication path for relays.		

Figure 62: SEC-BIG-15: Authenticating Relays

6.16 SEC-BIG-16: Relay Diagnostics Page

ID: SEC-BIG-16	Relay Diagnostics Page	Recommended
Detail: It is generally recommended to password protect the Relay Diagnostics Page. If it is not used, it should also be disabled.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Ensure the following values are set appropriately. o Relay → _BESRelay_Diagnostics_Password = (password hash) o Relay → _BESRelay_Diagnostics_Enable = 0		

Figure 63: SEC-BIG-16: Relay Diagnostics Page

6.17 SEC-BIG-17: Password Protection

ID: SEC-BIG-17	Password Protection	Recommended
Detail: The default password rules permit low complexity (e.g. a six character minimum). A more complex password rule and associated parameters should be defined based on the business standards.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Ensure the following values are aligned with business standards. o BigFix Administration Tool → passwordComplexityRegex o BigFix Administration Tool → passwordComplexityDescription o BigFix Administration Tool → passwordsRemembered o BigFix Administration Tool → maximumPasswordAgeDays o BigFix Administration Tool → minimumPasswordLength o BigFix Administration Tool → enforcePasswordComplexity o BigFix Administration Tool → accountLockoutThreshold o BigFix Administration Tool → accountLockoutDurationSeconds		

Figure 64: SEC-BIG-17: Password Protection

6.18 SEC-BIG-18: Login Protection

ID: SEC-BIG-18	Login Protection	Recommended
Detail: The timeouts associated with user authentication should be defined based on the business standards. The setting “timeoutLogoutMinutes” will close the console session after a defined period of time. It should be set to ensure console access is not abused. The login warning banner may also be modified.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Ensure the following values are aligned with business standards. o BigFix Administration Tool → loginTimeoutSeconds o BigFix Administration Tool → timeoutLockMinutes o BigFix Administration Tool → timeoutLogoutMinutes o BigFix Administration Tool → loginWarningBanner		

Figure 65: SEC-BIG-18: Login Protection

6.19 SEC-BIG-19: Action Confirmation

ID: SEC-BIG-19	Action Confirmation	Recommended
Detail: This option will force confirmation every time an action is taken. It provides a basic level of protection against user error.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Ensure the following values are set appropriately. o Console → requireConfirmAction = 1		

Figure 66: SEC-BIG-19: Action Confirmation

6.20 SEC-BIG-20: Agent Secure Registration

ID: SEC-BIG-20	Agent Secure Registration	Recommended
Detail: The _BESClient_SecureRegistration setting may be used in scenarios where manual key exchange is necessary, though manual entry is still preferred. The BigFix 10.0.5 release includes protocol improvements for the key exchange. This aligns with the SEC-BIG-09 recommendation to be on the most recent BigFix distribution.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: Manual key exchange per the product reference.		

Figure 67: SEC-BIG-20: Agent Secure Registration

6.21 SEC-BIG-21: Admin Key Protection

ID: SEC-BIG-21	Admin Key Protection	Recommended
Detail: The BES Admin Key should not be stored on the root server. It should be securely stored on an alternate node and should be guarded when in use.		
Verification: • Reference(s): BigFix Knowledge Center: URL • Procedure: This is an administrative function on top of BigFix and should be managed within the business guidelines.		

Figure 68: SEC-BIG-21: Admin Key Protection

.

REFERENCES

BigFix Knowledge Center

BigFix Resource Center

BigFix Capacity Planning Guide

BigFix Maintenance Guide

MX Performance Toolkit for BigFix

Blog: BigFix 10 Capacity Planning

Blog: BigFix 10 Infrastructure Monitoring

Blog: BigFix FillDB Performance

Blog: BigFix MLE Enablement

BigFix Performance Configurations

BigFix Common Criteria Certification

BigFix System Requirements

MS SQL Maximum Degree of Parallelism

MS SQL Cost Threshold for Parallelism

Performance Best Practices for VMware vSphere™ 5.5

Performance Best Practices for VMware vSphere™ 6.7

Performance Best Practices for VMware vSphere™ 7.0

Best practices for virtual machine snapshots in the VMware environment

VMware: Troubleshooting ESX/ESXi virtual machine performance issues

VMware: Troubleshooting virtual machine performance issues

VMware: Performance Blog

Notices

This information was developed for products and services offered in the U.S.A.

HCL may not offer the products, services, or features discussed in this document in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

HCL may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to HCL TECHNOLOGIES LIMITED email: products-info@hcl.com

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: HCL TECHNOLOGIES LIMITED PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HCL may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-HCL Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this HCL product and use of those Web sites is at your own risk.

HCL may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact HCL TECHNOLOGIES LIMITED email: products-info@hcl.com

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by HCL under terms of the HCL License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

All statements regarding HCL's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All HCL prices shown are HCL's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

HCL, and other HCL graphics, logos, and service names including "hcltech.com" are trademarks of HCL. Except as specifically permitted herein, these Trademarks may not be used without the prior written permission from HCL. All other trademarks not owned by HCL that appear on this website are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by HCL.

IBM and other IBM graphics, logos, products and services are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Oracle database, Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

VMware's and all VMWare trademarks and logos are trademarks or registered trademarks in the United States and certain other countries.

Dell, EMC, DellEMC and other trademarks are trademarks of Dell Inc. or its subsidiaries in the United States and certain other countries.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo and JBoss are registered trademarks of Red Hat, Inc. in the U.S. and other countries. Linux is a registered trademark of Linus Torvalds. All other trademarks are the property of their respective owners.

Mozilla and all Mozilla trademarks and logos are trademarks or registered trademarks in the United States and certain other countries.

Google LLC All rights reserved. Google and the Google Logo are registered trademarks of Google LLC.

NETAPP, the NETAPP logo, and the marks listed at www.netapp.com/TM are trademarks of NetApp, Inc.