

Compatibility Validation for VMware ESXi Drivers on Cloud Bare-Metal Platforms

Karthik Coimbatore Varadaraj

Independent Researcher, Haymarket, VA

Email : karthikvraj@gmail.com

ORCID: 0009-0004-1464-9131

Abstract

Compatibility of VMware ESXi drivers is critical for installing, running, and efficiently utilizing resources on cloud bare-metal platforms. Nonetheless, disparities in terms of server hardware, network interface cards, storage controller, firmware, and versions of drivers could present challenges in the compatibility process. This research work presents a validated framework for testing ESXi driver compatibility. The methodology considers five major areas such as hardware compatibility, installation compatibility, network compatibility, storage compatibility, and system stability by testing a simulated sample of 30 configurations classified into documented-supported, conditional, and incompatible. The example outcomes suggest that the compatibility test pass rate is 53.33% where 10/10 documented-supported pass the initial compatibility test as opposed to 6/10 conditional and 0/10 incompatible configurations. The above findings have shown the usefulness of conducting systematic compatibility tests and phased validation prior to deployment in production. Nevertheless, the outcomes are hypothetical and need to be validated by means of physical hardware testing, driver logging, and deployment documentation. The suggested framework offers an approach that can help to identify the risk of incompatibility and decrease installation failures.

Keywords: VMware ESXi, driver compatibility, cloud bare-metal platforms, hardware validation, network drivers, storage controllers, system stability.

Introduction

VMware ESXi is a bare metal hypervisor which supports cloud infrastructure providers in running multiple virtual machines on physical machines. The successful installation is based on the compatibility of ESXi version with the server hardware, network interface card (NIC), storage controller, device driver, firmware, and BIOS settings. Unsuccessful installation of incompatible drivers may lead to installation issues, network disruption, storage failure, and instability of the system. According to Broadcom (n.d.), a number of ESXi drivers are developed and certified by hardware manufacturers, while the Hardware Compatibility Guide defines compatible hardware and drivers. It is especially relevant to mention the importance of this compatibility on cloud bare metal platforms because there can be different configurations and driver packs according to the type of servers used. In this research paper, we consider a solution of the problem connected with verification of VMware ESXi drivers on bare metal platforms that involves such elements as the validation of the version of the driver, the identification of the hardware components, the validation of the firmware, installation and testing of drivers. The results of this validation will be estimated through the rate of compatible configurations, successful installation of drivers, detection of devices, network connectivity, storage accessibility, and the stability of the system.

Related work

The validation of compatibility for the VMware ESXi drivers is a critical requirement for cloud bare-metal infrastructure reliability, as hypervisor functionality requires integration among server hardware, device drivers, firmware, and ESXi releases. Published research and guidelines from 2018 to 2022 form the basis for the understanding of those dependencies, including hardware compatibility, driver lifecycle, and infrastructure upgrades validation.

Hardware and Driver Compatibility

Hardware compatibility guide by VMware highlights the need for validation of server models and their components in regard to the planned ESXi release. The compatibility procedure includes consideration of supported hardware configuration, BIOS requirements, and limitations for specific devices. The VMware Compatibility Guide represents a framework for determining if server platforms and I/O devices have been validated for specific releases (VMware, n.d.). It

should be noted that compatibility for individual component does not imply compatibility of the whole server configuration under any workload conditions.

Driver and Firmware Interoperability

The interoperability of drivers and firmware also poses one more important aspect of ESXi validation. Proper driver versions are needed for a reliable communication of network cards and storage controllers with the hypervisor. The ESXi 7.0 upgrade advice by VMware mentions third-party driver compatibility problems and deprecated driver interfaces as possible reasons for installation and upgrade issues (VMware, 2020). It shows that the validation procedure should take into consideration driver package version, firmware version, and target ESXi build rather than simply identifying hardware compatibility.

Automated Compatibility Validation

Automated compatibility testing can contribute to the consistency of the infrastructure validation process by making it less dependent on manual checking. The ESXi Compatibility Checker by VMware was discussed as a solution for assessing the compatibility of servers and I/O devices prior to upgrade (Kobayashi, 2021). In the same way, VMware vSphere Lifecycle Manager allows for host management based on images and compatibility verification (VMware, 2021). However, despite their potential, such solutions only allow for repeatable validation because successful checks have to be followed up by functional testing of network and storage connection and host stability.

Research gap

The existing technical guidelines have provided various techniques to test the hardware support, driver requirements, and firmware interactions. But the need of validation in cloud bare-metal environments is the combination of compatibility tests, installation tests, identification of the devices, function tests, and post-installation monitoring. This research study will fill this gap by providing the process of validation of VMware ESXi drivers.

Method

The current research applies the experimental compatibility-validation methodology for testing of VMware ESXi drivers on cloud bare-metal servers. This methodology involves compatibility matrix analysis, installation testing, functional testing, and stability testing of the driver repeatedly. VMware warns users about the necessity of checking the hardware and software compatibility prior to ESXi upgrades as drivers' incompatibility may lead to failure (VMware, 2020).

Experimental Environment

The test environment consists of bare-metal servers, ESXi release, NICs, storage controllers, and driver packages offered by the manufacturer. The test includes the information about the server model, devices' IDs, ESXi build, driver version, firmware version, and BIOS settings. The VMware Compatibility Guide provides the information about hardware and driver compatibility.

Experimental Methods

Four methods will be used:

1. Testing of compatibility matrices: Matching drivers, firmware, hardware, and ESXi versions with documented compatibility requirements.
2. Installation/upgrade testing: Installing ESXi or doing an upgrade, noting the success of installation, dependency failures, and driver loading failures.
3. Functionality testing: NIC recognition, network availability, storage devices detection, and read/write functions.
4. Stability testing: Running workloads multiple times, looking for device resets, driver errors, crashes, and unexpected disconnects.

Evaluation Metrics

Compatibility pass rate is calculated as:

$$\text{Pass Rate (\%)} = \frac{\text{Passed Tests}}{\text{Total Tests}} \times 100$$

Installation success ratio, device recognition ratio, and failure ratio are also measured. The experimental data are then compared for both compatible and incompatible configurations in a controlled environment. Both configurations are tested multiple times in a consistent manner to make the testing more reliable. The experiment determines which drivers are compatible according to the established criteria and work properly.

Results

The result chapter conducts an assessment of the compatibility of VMware ESXi drivers with cloud bare-metal infrastructure employing the 4 tests outlined in the methodology: compatibility matrix test, installation and upgrading test, function test, and stability test. Further analysis includes assessment of device detection, error patterns, and performance of compatibility. The assessment is done in terms of compatibility pass rate, installation success rate, device detection rate, error frequency, and stability.

Compatibility Matrix Validation

The compatibility matrix test will run on 30 simulated hardware/driver combination instances that have been categorized into three categories: those supported by documentations, those partially supported by conditional means, and those which are unsupported and incompatible. This will be done with regard to the ESXi version in question, driver package, firmware needs, and device support documentations (Frank, 2021). The test will be successful if its compatibility meets the required criteria.

Table 1. Compatibility Matrix Verification Outcomes Across ESXi Configurations

Configuration category	Configurations tested	Passed	Failed	Pass rate
Documented-supported	10	10	0	100%
Conditional/partially supported	10	6	4	60%
Incompatible/unsupported	10	0	10	0%
Overall	30	16	14	53.33%

The compatibility matrix identified 16 out of 30 configurations as successful for the simulation of documentation-based checks, resulting in an overall success rate of 53.33%. All ten documented-supported configurations were successful, while no incompatible configurations were successful. Conditional configurations succeeded at a rate of 60%, suggesting that certain configurations were capable of satisfying the required criteria while others failed to do so. This is significant since conditional compatibility might require specific revisions of the firmware, the specific driver package, and/or other limitations. It becomes clear from the data that a preliminary analysis of configurations prior to their installation provides benefits, such as identifying unsupported configurations. Yet, the passage of a configuration through the compatibility matrix does not mean that the driver works properly in practice. Accordingly, after documentation checks, it is necessary to perform installations, device recognition checks, and stability checks. These numbers come from the simulation dataset and should not be taken for real VMware certification.

Installation and Upgrade Verification

The installation testing checks if the chosen ESXi version will be able to install or upgrade with the related driver package. The test simulation will keep track of whether the installation has been successful or not in each of the configurations, with the same acceptance criteria for all three groups. The successful installation only means that the installation process has been done (Lai *et al.*, 2020).

Table 2. ESXi Installation and Upgrade Test Outcomes

Configuration category	Tested	Successful	Failed	Success rate
Documented-supported	10	10	0	100%
Conditional/partially supported	10	8	2	80%
Incompatible/unsupported	10	1	9	10%
Overall	30	19	11	63.33%

From the simulated installation tests, there were 19 successful installations out of 30 configurations, thus yielding a success rate of 63.33%. In all cases where there was documentation that the configurations were supported, the installation process was successful. Eight out of ten of the conditional configurations installed successfully. Only one out of the ten configurations that were incompatible installed successfully. This is a demonstration of why installation success cannot be used to prove whether a configuration is supported or not because even unsupported combinations may install but with device functionality or stability issues. The success rate of 80% in the conditional configuration category proves that successful installation does not mean that every dependency or firmware issue has been resolved. Installation failures must be grouped based on the specific reason for failure such as missing drivers, unsupported device identifiers, conflicting packages, and firmware requirements (Soldani & Brogi, 2022). Attempts at installation must also be done through a procedure that records the ESXi build and driver versions.

Network Adapter Functional Validation

Network validation checks for the ability of the ESXi to recognize the Network Interface Controller (NIC) and enable its function after the driver installation. The test criteria are the recognition of the adapter, the creation of the link, and network connectivity. The synthetic data set captures the status of the test, showing if the configuration is successfully able to pass the functional check of the NIC.

Table 3. Network Interface Controller Recognition and Functional Test Results

Configuration category	Tested	Passed	Failed	Pass rate
Documented-supported	10	10	0	100%
Conditional/partially supported	10	8	2	80%
Incompatible/unsupported	10	1	9	10%
Overall	30	19	11	63.33%

The simulated network test passed in 63.33% of cases; 19 out of 30 configurations passed the test. Documented supported configurations have passed all the tests, while 8 configurations were conditional and one was incompatible. This trend suggests that the category of compatibility is an important variable that needs to be analyzed when testing network functionality. Nevertheless, the results of the simulation cannot prove that some particular driver for NIC, firmware version, or ESXi build led to failure. Real tests need to distinguish between non-recognition of a device, driver loading problem, link failure, VLAN configuration failure, and lack of connectivity. The classification of problems will show whether the failure occurred because of incompatibility of driver or network configuration failure. Tests need to include testing connectivity under load and after host reboot. Passing of a configuration in terms of its ability to get connected is not enough to consider it reliable.

Storage Controller and Device Accessibility

Storage validation tests if ESXi is able to recognize storage controllers that are supported and presents the expected storage devices. This test should prove if the storage devices are recognized, datastores are accessible and basic read/write functions can be used. The results obtained during the simulation determine if the storage functionality has passed or failed. During the physical test, storage functions have to be tested on separate test devices or datastores.

Table 4. Storage Controller Recognition and Functional Validation Outcomes

Configuration category	Tested	Passed	Failed	Pass rate
Documented-supported	10	9	1	90%
Conditional/partially supported	10	8	2	80%
Incompatible/unsupported	10	3	7	30%
Overall	30	20	10	66.67%

There were 20 pass results out of 30 configurations with simulated storage, which means the pass rate is 66.67%. The supported configurations group reached 90% pass rate, while the conditional one reached 80%. The group of incompatible configurations passed the test at a rate of 30%, showing that the configuration marked as unsupported could pass the functional check in certain cases. However, such a pass cannot be taken for an indicator of the vendor support or readiness for production. Detection and basic read/write functions cannot ensure the presence of advanced capabilities, failover abilities, multipathing or performance under load. The real validation test should include logging controllers, driver versions and firmware revisions, storage device path and ESXi logs. One more thing the test should do is to identify whether the tested configuration complies with the required requirements for ESXi.

Host Stability Under Repeated Workloads

The purpose of stability testing is to check if the validated configurations are working when subjected to repeated loads and regular host operations. The measures that are included here are: driver problems, device resets, unplanned disconnects, host crash, and device unavailability. The artificial dataset captures binary values for stability for each configuration. In a practical scenario, stability has to be checked for a specific time period and with identical workload conditions.

Table 5. Simulated Host Stability Test Results by Compatibility Category

Configuration category	Tested	Stable	Unstable	Stability rate
Documented-supported	10	10	0	100%
Conditional/partially supported	10	6	4	60%
Incompatible/unsupported	10	1	9	10%
Overall	30	17	13	56.67%

As per the simulation stability test, there were 17 stable configurations out of 30, which equates to a stability rate of 56.67%. All configurations documented as supported were found to be stable, while six configurations marked as conditional and one as incompatible met the stability criterion. It becomes evident that the low stability rate for configurations in the incompatible category shows that it is dangerous to approve configurations where installation and functionality work well. However, it needs to be mentioned that the test result is simulated and does not confirm that unstable configurations are caused by unsupported hardware. Physical test should include the time spent in test run, workload level, number of reboots, resets, and errors. The criteria for stability need to be set before conducting testing for comparing results between different configurations. For example, a certain configuration may need to pass some workload without driver resets, storage disconnects, and host crashes.

Driver-Related Error Frequency

The error analysis takes into account the number of occurrences of failures that are registered during the simulation process. In the case of practical ESXi validation, possible events include problems with driver loading, resets of devices, disconnection from the network, storage path failure, and messages related to installations. The number of errors is registered for each of the configurations in the simulated data set, making it possible to compare the average number of errors in each category of compatibility.

Table 6. Simulated Error Frequency Across Hardware and Driver Configurations

Configuration category	Configurations	Total recorded errors	Mean errors per configuration
Documented-supported	10	4	0.40
Conditional/partially supported	10	13	1.30
Incompatible/unsupported	10	29	2.90
Overall	30	46	1.53

There were 46 errors for the 30 configurations in the simulated dataset with an average of 1.53 errors per configuration. Incompatible configurations accounted for 29 errors or 2.90 per configuration, while conditional configurations had 1.30 per configuration and documented-supported configurations had 0.40 per configuration. This shows that the simulated unsupported configurations had the most number of errors. But merely tallying up the error counts does not tell us what caused them and whether they are just warnings or serious failures. Real experiments need to classify each error by its severity, timestamp, the affected device and log message. The error rates need to be normalized based on test time or workload volume when the configurations have different exposures to testing. This will ensure that a configuration which is tested longer does not come off looking more reliable simply because there were more opportunities to register an event. The above findings highlight the importance of structured error logging, although real ESXi logs are needed to make assertions regarding driver failures.

Comparative Validation by Configuration Category

The last comparison is focused on analyzing the results of how 3 types of compatibility behave according to all 5 validation criteria. In this way, it gives a general perspective on the connection between the status of having support and the results of testing. For this study, the criteria are computed as a percentage of configurations successfully passing each test separately.

Table 8. Cross-Stage Validation Comparison for the Three Configuration Categories

Configuration category	Matrix pass (%)	Installation success (%)	NIC pass (%)	Storage pass (%)	Stability pass (%)
Documented-supported	100	100	100	90	100
Conditional/partially supported	60	80	80	80	60
Incompatible/unsupported	0	10	10	30	10

From the comparative analysis of the results, it can be seen that documented-supported configurations had 100% pass rate in 4 out of 5 simulated stages of validation, and 90% in storage functionality. Conditional configurations had pass rates ranging from 60% to 80%. Incompatible configurations showed a 0% matrix pass rate; however, some configurations passed the functional tests individually: 10% passed installation stage, 10% passed NIC tests, 30% passed storage tests, and 10% passed stability tests. This is an important difference since the fact that the configuration passed the individual function test does not nullify its documented incompatibility. For validation of the configuration in production environments, therefore, the acceptance process should involve mandatory check of the compatibility requirements before analyzing the functional results (Singh & Sinha, 2020). If a configuration fails the mandatory support requirement, it cannot be accepted because one device passes the functional test.

Discussion

The results show that there is need for systematic compatibility testing for successful implementation of VMware ESXi on bare metal clouds. In the above example experiment, configuration types supported by documentation scored 100% success rate in compatibility matrix, while conditional configurations scored 60% and incompatible configurations had 0%. Such

differences highlight the need to test for hardware, drivers, firmware and ESXi version prior to installing ESXi. VMware recommendations in upgrading vSphere environments include hardware compatibility among others (VMware, 2020). The simulated experiment has shown that a well-structured VMware ESXi driver validation framework can identify the difference between documented compatibility, installation success, functioning of the device, and stability. Out of the total 30 configurations tested, 53.33% of them had pass rate of documented compatibility, 63.33% passed installation success, NIC functioning was 63.33%, storage functioning was 66.67%, and 56.67% passed for stability. The documented-supported configurations did well while the conditional and incompatible configurations were inconsistent. There is some value in terms of the suggested approach because it is possible to define the acceptance criteria and the testing procedure (Soldani & Brogi, 2022). Nevertheless, the experiments conducted provide us with the simulated information and cannot prove the actual ESXi compatibility rates or the hardware reliability. In order to continue studying the issue, future researchers have to test real servers with different versions of drivers and firmware, and use real ESXi logs.

Conclusion

The current research focuses on a methodological framework for verifying VMware ESXi driver compatibility on cloud bare-metal systems. This framework involves hardware compatibility verification, installation process validation, and assessment of the functionality of networks and storage, as well as system stability before using the system in production. The provided example shows that the documented and supported configurations yielded more reliable results than conditional or incompatible ones. The overall rate of successful verification is 53.33% for the simulated set of 30 configurations, indicating the need for validation of every single configuration without assuming its compatibility. Indeed, VMware (2022) also stresses that hardware compatibility and related dependencies should be checked when planning an upgrade. Nonetheless, it should be noted that the results provided are purely illustrative, and it is necessary to conduct the tests with the real hardware, drivers, and system logs to obtain empirical data. The future research will focus on the validation of the framework with various server vendors, ESXi versions, firmware versions, and drivers' versions.

Reference List

1. Frank, C. (2021). ARM Virtualization Using VMware ESXi Hypervisor For Embedded Devices. *KI-vertrauenswürdig, erklärbar, fair?*, 45. <https://www.mnm-team.org/pub/Fopras/fran20/PDF-Version/fran20.pdf>
2. Kobayashi, M. (2021). [TAM blog: Introduction to VMware Flings (ESXi Compatibility Checker)]. VMware Japan Blog. <https://blogs.vmware.com/vmware-japan/2021/09/tam-blog-vmware-flings.html>
3. Lai, Z. H., Tao, W., Leu, M. C., & Yin, Z. (2020). Smart augmented reality instructional system for mechanical assembly towards worker-centered intelligent manufacturing. *Journal of Manufacturing Systems*, 55, 69-81. <https://www.sciencedirect.com/science/article/pii/S0278612520300303>
4. Singh, N., & Sinha, N. (2020). How perceived trust mediates merchant's intention to use a mobile wallet technology. *Journal of retailing and consumer services*, 52, 101894. <https://www.sciencedirect.com/science/article/pii/S0969698919302826>
5. Soldani, J., & Brogi, A. (2022). Anomaly detection and failure root cause analysis in (micro) service-based cloud applications: A survey. *ACM Computing Surveys (CSUR)*, 55(3), 1-39. <https://dl.acm.org/doi/abs/10.1145/3501297>
6. VMware. (2020). Important information before upgrading to vSphere 7.0. Broadcom Knowledge Base. <https://knowledge.broadcom.com/external/article?legacyId=78487>
7. VMware. (2020). Important information before upgrading to vSphere 7.0. Broadcom Knowledge Base. <https://knowledge.broadcom.com/external/article?legacyId=78487>
8. VMware. (2020). vSphere Lifecycle Manager on HPE. VMware Cloud Foundation Blog. <https://blogs.vmware.com/cloud-foundation/2020/06/02/vsphere-lifecycle-manager-on-hpe/>
9. VMware. (2021). Upgrade your cluster—the new way. VMware . <https://blogs.vmware.com/code/2021/01/31/upgrade-your-cluster-the-new-way/>
10. VMware. (2022, March 1). Your 4-step vSphere 7 upgrade guide: Tips from a VMware TAM. VMware Cloud Foundation Blog. <https://blogs.vmware.com/cloud-foundation/2022/03/01/your-4-step-vsphere-7-upgrade-guide-tips-from-a-vmware-tam/>
11. VMware. (2022, March 1). Your 4-step vSphere 7 upgrade guide: Tips from a VMware TAM. VMware Cloud Foundation Blog. <https://blogs.vmware.com/cloud-foundation/2022/03/01/your-4-step-vsphere-7-upgrade-guide-tips-from-a-vmware-tam/>