



Hewlett Packard
Enterprise

Helion OpenStack Carrier Grade 4.0

INTRODUCTION

Copyright Notice

© Copyright 2016 Hewlett Packard Enterprise Development LP

The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Links to third-party websites take you outside the Hewlett Packard Enterprise website. Hewlett Packard Enterprise has no control over and is not responsible for information outside the Hewlett Packard Enterprise website.

Acknowledgements

Java® and Oracle® are registered trademarks of Oracle and/or its affiliates

<http://www.hpe.com/info/storagewarranty>

Titanium Server

Cloud Administration, 16.10

Contents

1 Introduction to Titanium Server	1
Titanium Server Overview	1
Titanium Server Documentation	2
2 Core Titanium Server Capabilities	5
Overview of Core Titanium Server Capabilities	5
High Availability	6
High Availability	6
Controller Nodes and High Availability	7
Networking	8
High-performance Networking	8
Guest VLAN Support	9
Port-based DNAT Access for VMs	10
SR-IOV Ethernet Interfaces	10
PCI Passthrough Ethernet Interfaces	10
SR-IOV Encryption Acceleration	10
AVS Packet Tracing	11
Titanium Server and SDN	11
Improved VM Execution Environment	11
Deployment Options	12
Deployment Options	12
Titanium Server CPE Solution	13
Titanium Server Standard Solution	13
Hardware Management	13
Integrated Installation and Commissioning	13
Inventory Management Facilities	13
Hardware Profiles	14
UEFI Support	15
OAM Network	15
OAM Network Firewall	15
Alarm Generation and Reporting	16
REST API Support	16
Web Administration Interface Improvements	16
Centralized Log Collection and Analysis	17
Performance Management Extensions	18

Overview of Remote CLI SDK Module	18
Software Management	19
Patching Facilities	19
Backup and Restore Facilities	19
Upgrade Capabilities	19
Patch Orchestration Capabilities	20
Integrated OpenStack Services	20
Ceph-backed VM Storage	21
Heat	21
Heat Extensions	21
Resource Scaling (Autoscaling)	22
Titanium Server SDK	22
3 Supported OpenStack Features	25
Supported Openstack Features	25
4 Virtual Applications	35
Virtual Applications	35

Introduction to Titanium Server

[Titanium Server Overview](#) 1

[Titanium Server Documentation](#) 2

Titanium Server Overview

Wind River Titanium Server is a Network Functions Virtualization cloud solution capable of meeting demanding customer requirements for scalability, performance, capacity, and availability. It is not a one-size-fits-all solution, but a highly flexible framework that supports a range of deployment models, types and quantities of hardware, and tunable parameters to enable an optimized deployment for varying customer requirements.

Titanium Server brings together the flexibility and scalability of the IT cloud and the high availability and performance demanded by the Telecommunications industry to deliver a unique carrier-grade, industry-leading solution at a price-performance ratio well above alternative solutions. Titanium Server is aligned with the ETSI-NFV architecture.

This document provides guidelines, rules, and system parameters to assist cloud architects, installers, and administrators in deploying and scaling the Titanium Server system to meet application-specific requirements. Failure to follow these recommendations, guidelines and engineering rules will result in unknown performance trade-offs likely compromising reliability and performance.

Titanium Server Documentation

The Titanium Server documentation has been organized to help you locate information for specific types of activities, such as installation, administration, and VNF integration.

Table 1 **Titanium Server Documentation**

Document	Description
Wind River Introduction to Titanium Server	This document gives an introduction to Titanium Server and provides an overview of system capabilities, information on planning, and recommended workflows. Planning also helps ensure that the requirements of your hosted applications can be met, and the requirements of your Cloud administration and operations team can be met. It also ensures proper integration of Titanium Server into the target Data Center or Telecom Office, and helps you plan up front for future cloud growth.
Wind River Titanium Server Planning	This helps you plan out your Titanium Server installation, ensuring that you are fully prepared once you start your installation and configuration.
Wind River Titanium Server Installation for CPE Systems	This document provides information and instructions for installing Titanium Server CPE.
Wind River Titanium Server Installation for Systems with Controller Storage	This document provides information and instructions for installing Titanium Server configurations that are initially deployed using LVM-backed block storage on controller nodes.
Wind River Titanium Server Installation for Systems with Dedicated Storage	This document provides information and instructions for installing Titanium Server configurations that that are initially deployed using Ceph-backed block storage on dedicated storage nodes.
Wind River Titanium Server System Administration	This provides information pertaining to ongoing administration of a Titanium Server system, including information for managing the physical nodes and physical networks.

Document	Description
Wind River Titanium Server Software Management	This document provides instructions for applying patches and software upgrades to Titanium Server hosts.
Wind River Titanium Server Cloud Administration	This guide provides information on topics that an OpenStack administrator would be responsible for, except for the management of the physical nodes and physical networks, which is covered in the System Administration guide.
Wind River Titanium Server Tenant User's Guide	This provides information about the operational actions that a tenant user can take.
Wind River Titanium Server VNF Integration	This guide provides information to help you integrate your VNFs into a Titanium Server system.
Wind River Titanium Server for Regions	This provides installation and configuration information for deploying Titanium Server in any of the supported Regions configurations.
Wind River Titanium Server Software Development Kit	The Titanium Server Software Development Kit (SDK) provides drivers, daemons, API libraries, and configuration files that you can include in a guest image to leverage the extended capabilities of Titanium Server. These components can be used to enhance or extend the networking features of the applications and to access the virtual machine (VM) management capabilities of Titanium Server.
Wind River Titanium Server System Engineering Guidelines	This document provides engineering guidelines, rules, and system parameters to assist cloud architects, installers, and administrators in planning, deploying and scaling the Titanium Server system.
Wind River Titanium Server Software Defined Networking	This document provides information for using an SDN controller to manage Neutron services in Titanium Server.
Wind River Titanium Server Release Notes	These include high level details of new features in the current release, as well as information about known anomalies or usage caveats.

2

Core Titanium Server Capabilities

Overview of Core Titanium Server Capabilities	5
High Availability	6
Networking	8
Improved VM Execution Environment	11
Deployment Options	12
Hardware Management	13
OAM Network	15
Software Management	19
Integrated OpenStack Services	20
Ceph-backed VM Storage	21
Heat	21
Titanium Server SDK	22

Overview of Core Titanium Server Capabilities

Titanium Server optimizes carrier-grade technologies, Intel DPDK high-performance packet processing, open architectures, and the OpenStack software suite to implement a unique carrier-grade, high-availability architecture on which high-performance production systems can be deployed.

Key capabilities of OpenStack are incorporated, and improvements and extensions are introduced for flexibility and ease of use. For a complete list of supported Openstack capabilities, see [Supported Openstack Features](#) on page 25.

The following sections describe the features and capabilities that differentiate Titanium Server from other platforms.

High Availability

High Availability

Titanium Server provides a number of features to support highly available hosting of virtual machines.

Titanium Server includes the following extensions to OpenStack to support high availability:

1:1 OpenStack controller services

Automatic configuration of the OpenStack controller services in 1:1 active/standby mode across two controller nodes.

Fast detection of compute host failures

Implemented by using a highly efficient and highly scalable heartbeat protocol between the controller and compute nodes.

Fast recovery of virtual machines instances upon detection of a compute node failure

Extensions to Nova services that automatically re-schedule impacted virtual machine instances to available alternative compute nodes in the cluster.

Fast recovery of tenant network services upon detection of a compute node failure

Extensions to Neutron that automatically re-schedule impacted network services such as DHCP, L3 Routing, and User/Meta data server, for all affected tenant networks. This covers tenant networks spanning multiple compute nodes.

Fast and enhanced detection of virtual machines failures

Failure of any KVM/QEMU instance is automatically detected and reported by a modified nova-compute service, and recovery is automatically handled by the VIM.

Additionally, modified guests can make use of the Titanium Server Guest Heartbeat Library to register application-specific health check callbacks with the virtual machine. This registration allows the compute node to monitor the health of the guest application. The frequency of the health checks is determined at registration time. The semantics of the health check, that is, determining when the application is in a good or a bad state, is under the control of the application itself.

Automatic recovery of failed virtual machine instances

Extensions to Nova that automatically re-start a failed virtual machine on the same compute node. If the re-start operation fails, the virtual machine is re-scheduled to use an alternative compute node.

Enhanced virtual machine server groups

Titanium Server enhances virtual machine server groups by adding the following attributes:

- maximum number of virtual machine instances
- best effort or strict affinity policy

Virtual machines in the same server group can also use a Titanium Server server group messaging API for low-bandwidth communications.

Live migration support with DPDK-accelerated networking

Live migration support for virtual machines, using the high-performance networking options included with Titanium Server.

Graceful shutdown (and other operations) of virtual machines

Nova extensions that turn the default shutdown operation of virtual machines into an ACPI shutdown. Guest applications can therefore register shutdown scripts using standard ACPI mechanisms to execute operations such as closing files, updating persistent databases, or cleanly disconnecting from subscribed services.

Optionally, guest applications can use the Titanium Server Guest Heartbeat Library to register to receive process management requests such as shutdown, live-migrate, pause, auto-scale, and others. The guest applications can then reject the requests based upon application-specific state directives, or prepare for a graceful execution.

Link Aggregation (LAG) support

Support for LAG (with LACP), also known as Aggregate Ethernet, on controller and compute nodes for link protection.

Protected HA Middleware

Titanium Server Service Manager protects all critical processes. In the event of a process failure, individual processes can be independently restarted.

For hardware recovery, Titanium Server supports both single-fault and multiple-fault scenarios.

- If a single node fails, Titanium Server detects the fault and immediately initiates recovery, including VM evacuation if the fault is on a compute host.
- If multiple nodes fail, for example due to a rack power outage, Titanium Server enters *Multi Node Recovery Mode*. In this mode, the system is allowed to stabilize before VM recovery is attempted, minimizing the risk of thrashing. Administrators are advised to suspend manual activity on the system during this period. Error messages and customer log entries are provided as notification that the system has entered or exited Multi Node Recovery Mode.

Controller Nodes and High Availability

Services in the controller nodes run constantly in active/standby mode to provide continuity in the event of a controller failure.

Controller services are organized internally into the following groups:

Table 2 **Controller Service Groups**

Group	Description
Cloud Services	The enhanced OpenStack components, including Nova, Neutron, Cinder, Ceilometer, and Heat
Controller Services	Core Titanium Server services such as maintenance and inventory
Directory Services	LDAP services

Group	Description
OAM Services	OAM access services
Patching Services	Patching alarm services
Storage Monitoring Services	Storage alarm services
Storage Services	Storage REST API services
Web Services	The Titanium Server OpenStack Horizon service and web server

Each of these groups is run in 1:1 HA mode by the controllers. This means that while some service groups can be active on controller-0, and in standby on controller-1, others are active on controller-1, and in standby on controller-0.

The high-availability framework constantly monitors and reports on the health of the individual services within each of the service groups on both controllers. When a service fails, a decision is made on whether to restart it on the same controller, or to switch the corresponding service group to the other controller. This decision depends on the criticality and the dependencies of the affected service.

For maintenance purposes, when one of the controller nodes needs to be powered down for service, it is necessary to force all currently active service groups in one controller to switch to the other. This can be done from the **Hosts** tab on the Host Inventory page (available from **Admin > Platform > Host Inventory** in the left-hand pane) by selecting the option **swact** (switch active) in the **More** menu of the controller you want to take out of service.

The Active Controller

Services in the **Controller Services** group drive core functionality in Titanium Server. The controller where they are running is referred to as the *active* controller. The **Hosts** tab on the Host Inventory page, available from of the **Admin** panel lists the status of all hosts in the cluster; it reports the active controller as having the *Controller-Active* personality.

When working from the CLI on a controller node it is often important to ensure that you are working on the active controller, for example, to execute OpenStack **admin** operations, or to change the password of the **wrsroot** user account. For further details on the **wrsroot** account, see *Titanium Server System Administration: Linux User Accounts*.

You can ensure you are working on the active controller by using the OAM floating IP address as the destination address in the SSH command.

Networking

High-performance Networking

Titanium Server provides significantly improved network performance over the default open source OpenStack solution.

At the center of Titanium Server Networking is a DPDK-Accelerated Virtual L2 Switch (AVS), running on the compute node hosts. It provides connectivity between virtual machines on the

same or different compute nodes, and between virtual machines and external networks. AVS supports a variety of network connectivity options with the hosted virtual machines:

- Unmodified guests can use Linux networking and virtio drivers. This provides a mechanism to bring existing applications into the production environment immediately.

For virtio interfaces, Titanium Server supports **vhost-user** transparently by default. This allows QEMU and AVS to share virtio queues through shared memory, resulting in improved performance over standard virtio.

- For backward compatibility, Accelerated Virtual Port (AVP-KMOD) drivers are also supported.
- For the highest performance, guest applications can be modified to make use of Intel DPDK libraries and open-source AVP-PMD poll-mode drivers.

In addition to AVS, Titanium Server incorporates DPDK-Accelerated Neutron Virtual Router L3 Forwarding (AVR). Accelerated forwarding is used for directly attached tenant networks and subnets, as well as for gateway, SNAT, DNAT, and floating IP functionality.

Titanium Server also supports direct guest access to NICs using PCI passthrough or SR-IOV, with enhanced NUMA scheduling options compared to standard OpenStack.

For further performance improvements, Titanium Server supports direct access to PCI-based hardware accelerators, such as the Coletto Creek encryption accelerator from Intel. Titanium Server manages the allocation of SR-IOV VFs to VMs, and provides intelligent scheduling to optimize NUMA node affinity.

For more information, see *Titanium Server Planning: Ethernet Interface Configuration*

High Performance Networking Drivers

The Titanium Server SDK provides high-performance AVS-compatible networking drivers, including accelerated kernel network drivers and accelerated DPDK network drivers.

Guest VLAN Support

Titanium Server guests can make use of Titanium Server-defined VLANs to encapsulate IP traffic from a single or multiple IP subnets on a virtual Ethernet interface.

Titanium Server tenants can define one or more VLAN-tagged IP subnets on a single tenant network to allow their guests' traffic to be encapsulated on VLAN IDs of their choice. These Titanium Server-defined VLAN-tagged IP subnets have access to all of the services of the virtualized network infrastructure, such as DHCP, virtual routing, meta-data server, and so on.

Alternately, Titanium Server guests can make use of transparent VLANs, in which packets are encapsulated within a provider network segment without removing or modifying the guest VLAN tag(s). VLAN Transparent provides more flexibility with respect to how VLAN tagged packets are handled without requiring that you pre-define which VLAN instances will be used. A guest can send/receive VLAN tagged packets (802.1q) and/or double VLAN tagged packets (802.1ad) without first defining a VLAN-tagged subnet in Titanium Server. In addition, a VLAN transparent network will also propagate VLAN priority information (802.1p) to the destination VM instance. However, as a consequence of allowing arbitrary VLAN instances, you lose the ability to access DHCP servers, virtual routers, or meta-data servers over those VLAN-tagged networks.

Support for Guest VLANs is a key capability that facilitates the porting of current user applications running on dedicated physical servers using VLANs to a virtualized environment.

Port-based DNAT Access for VMs

Titanium Server supports port-based DNAT, or port forwarding, for virtual routers on which SNAT is enabled.

This provides for externally initiated connections to multiple VMs using a single external IP address. For more information, see *Titanium Server Tenant User's Guide: Configuring Port-based DNAT on a Virtual Router*.

SR-IOV Ethernet Interfaces

A SR-IOV Ethernet interface is a physical PCI Ethernet NIC that implements hardware-based virtualization mechanisms to expose multiple virtual network interfaces that can be used by one or more virtual machines simultaneously.

The PCI-SIG Single Root I/O Virtualization and Sharing (SR-IOV) specification defines a standardized mechanism to create individual virtual Ethernet devices from a single physical Ethernet interface. For each exposed virtual Ethernet device, formally referred to as a *Virtual Function* (VF), the SR-IOV interface provides separate management memory space, work queues, interrupts resources, and DMA streams, while utilizing common resources behind the host interface. Each VF therefore has direct access to the hardware and can be considered to be an independent Ethernet interface.

PCI Passthrough Ethernet Interfaces

A passthrough Ethernet interface is a physical PCI Ethernet NIC on a compute node to which a virtual machine is granted direct access.

This minimizes packet processing delays but at the same time demands special operational considerations.

For all purposes, a PCI passthrough interface behaves as if it were physically attached to the virtual machine. Therefore, any potential throughput limitations coming from the virtualized environment, such as the ones introduced by internal copying of data buffers, are eliminated. However, by bypassing the virtualized environment, the use of PCI passthrough Ethernet devices introduces several restrictions that you must take into consideration. They include:

- no support for LAG, QoS, ACL, or host interface monitoring
- no support for live migration
- no access to the compute node's AVS switch

SR-IOV Encryption Acceleration

Titanium Server supports PCI SR-IOV access for encryption acceleration.

Titanium Server supports SR-IOV access for the Intel AV-ICE02 VPN Acceleration Card, based on the Intel Coletto Creek 8925/8950 chipset with QuickAssist™ technology. (Due to driver limitations, PCI passthrough access is not currently supported.) If this card is present on an available host, you can provide VMs with access to one or more SR-IOV devices to improve performance for encrypted communications.



CAUTION: Live migration is not supported for instances using SR-IOV devices.

To expose the device to VMs, see *Titanium Server System Administration: Exposing a Device for Use by VMs*. To provide a VM with access to this device, see *Titanium Server Cloud Administration: Accessing a PCI Device from a VM*.



NOTE: To use PCI passthrough or SR-IOV devices, you must have Intel VT-x and Intel VT-d features enabled in the BIOS.

AVS Packet Tracing

Titanium Server provides a live packet-trace capture utility for vSwitch logical interfaces, using tcpdump.

This utility is intended only for development or other uses where performance degradation and potential packet loss are not a concern. For more information, see *Titanium Server System Administration: Performing Packet Tracing on vSwitch Interfaces*.

Titanium Server and SDN

Titanium Server is interoperable with OpenFlow 1.3 for integration in data centers that use software defined networking (SDN) as the networking backbone.

Titanium Server can be configured to use SDN as an alternate backend for Neutron networking. An SDN-specific Neutron plugin uses SDN controller northbound APIs to send Neutron tenant networking configurations to an OpenStack / Neutron integration application running on a remote SDN controller. The SDN controller implements the tenant networking by using OSVDB and OpenFlow interfaces to send switch configuration and flow rules to the accelerated vSwitches (AVS) on Titanium Server compute nodes.

When enabled, the SDN controller manages Neutron L2 services and optionally L3 services. If the option for the SDN controller to manage L3 services is not enabled, then L3 services are managed locally.

This feature is provided for demonstration purposes. For more about Titanium Server and SDN, refer to the *Titanium Server Release Notes*.



NOTE: SDN interoperability is configurable during system installation only. It cannot be added or removed after installation.

Improved VM Execution Environment

Titanium Server provides an improved execution environment for better and more predictable virtual machine performance.

Enhancements to the execution environment include:

- Platform resource utilization monitoring, to ensure that VMs can be scheduled on a host only if the host OS is using no more than 80% of the CPU or memory allotted for platform use.
- Low-latency, bounded delivery of virtual interrupts, and availability of low-latency high resolution timers. This is done by leveraging the optimized kernel and KVM/QEMU implementations of the Wind River Open Virtualization Platform (OVP).
- Option to allocate 2 MB or 1 GB huge memory pages to reduce swapping and TLB lookup misses.
- Improved resource tracking of VMs that use dedicated CPUs, to ensure successful migration and evacuation.
- Support for affining guest NUMA nodes to specific host NUMA nodes.
- Capability to specify specific CPU models to be used by a virtual machine in order to leverage advanced features of the CPU architecture.
- Support for PCI passthrough and SR-IOV access to Ethernet interfaces
- Support for SR-IOV access to PCI-based encryption acceleration
- Improved performance for standard virtio NICs using vhost-user, with optional multi-queue support for virtio interfaces
- Option to use cached RAW images for accelerated Cinder volume creation
- Option to use thin provisioning on all controller-based LVM Cinder volumes for faster deletion and secure deletion support, or thick provisioning for faster volume creation and faster initial disk writes
- Copy-on-write (CoW) image-based local ephemeral storage backing for instance launch and delete optimization

Deployment Options

Deployment Options

Titanium Server presents several deployment options.

These deployment options include:

- Titanium Server Standard Solutions
 - Titanium Server Standard Configuration with Dedicated Storage
 - Titanium Server Standard Configuration with Controller Storage
 - Titanium Server in Multi-Region Environment
- Titanium Server CPE

Titanium Server CPE Solution

Titanium Server can be deployed on a system consisting of just two nodes, one active and one redundant.

In the Titanium Server CPE (Customer Premises Equipment) solution, each node implements controller, storage, and compute functions, and provides virtual machine hosting. This provides an ideal system for consolidating a small set of standalone server-based products, which may have various operating systems, networking requirements, and so on, into a simple and compact physical solution that offers ease of deployment, high availability, resource autoscaling, and other benefits of a cloud implementation.

Titanium Server Standard Solution

Titanium Server can be deployed as a standard solution with either controller storage or dedicated storage.

The Titanium Server standard solution includes options for controller storage and dedicated storage. Titanium Server can also be deployed in Regions configurations.

Hardware Management

Integrated Installation and Commissioning

Titanium Server substantially simplifies and integrates the installation and commissioning sequence.

Software for the initial controller node is installed from a USB flash drive or PXE boot server, and configured for operation with a single script. Software for subsequent hosts is installed over the internal management network from the initial controller node, and configured using either the web administration interface, or the command line on the controller.

This is in contrast with the installation and commissioning sequence of the open-source OpenStack, which involves the setup of numerous individual configuration files, the installation of various backends supporting OpenStack components, and the configuration of multiple Linux services and other open-source programs.

Titanium Server also supports automated installation and commissioning using response files for initial configuration, and manifest files for bulk installation of subsequent hosts.

Inventory Management Facilities

Titanium Server provides complete inventory management of hosts in the OpenStack cloud, allowing the system administrator to install, configure, and maintain the individual servers.

The inventory service gives the system administrator the following capabilities:

- Host management

- discovery of new hosts in the cluster
- installation of the appropriate load, controller, compute, or storage images
- configuration of the management, OAM, infrastructure, and provider network interfaces on each host
- creation and use of profiles for CPU, interface, memory, and storage assignments to simplify host configuration
- configuration of the number of CPUs allocated to the Accelerated Virtual Switch (AVS) on each compute node
- configuration of huge memory page allocations for VM use
- Administrative operations
 - lock/unlock
 - switch active controller host or service
 - reboot and reset host
 - power-on and power-off
 - software re-installation
- Status reporting
 - admin state
 - operational state
 - availability state
 - uptime
 - real-time command execution reports, such as booting and testing
 - hardware sensors (for example, temperature or voltage sensors)
 - Centralized visual reporting for compute node data interfaces and provider network topologies, including alarm status indicators
- Host resources
 - processor, sockets, cores
 - memory
 - disk
 - network interfaces
 - additional hardware devices, such as cryptographic and compression devices

Hardware Profiles

You can capture aspects of a host configuration as a *hardware profile*, and then use the profile to apply the configuration to other hosts.

You can capture the following aspects of a host configuration:

- CPU assignments for platform, vSwitch, or VM use (see *Titanium Server Installation: CPU Profiles*)
- Ethernet port and interface attachments (see *Titanium Server Installation: Interface Profiles*)
- Storage resource allocations (see *Titanium Server Installation: Storage Profiles*)



NOTE: Storage profiles for compute-based or CPE ephemeral storage (that is, storage profiles containing volume group and physical volume information) can be applied in two scenarios:

- on initial installation where a nova-local volume group has not been previously provisioned
- on a previously provisioned host where the nova-local volume group has been marked for removal

On a previously provisioned host, delete the nova-local volume group prior to applying the profile.

- Memory allocations for platform and VM use (see *Titanium Server Installation: Memory Profiles*)

You can create profiles from existing hosts (see *Titanium Server Installation: Creating Hardware Profiles from an Existing Host*), or define them using XML and then import them to Titanium Server (see *Titanium Server Installation: Importing Hardware Profiles*.)

To apply profiles to hosts, see *Titanium Server Installation: Applying Hardware Profiles*.

If you prefer, you can work with profiles using the CLI. For more information, see *Titanium Server Installation: Managing Hardware Profiles Using the CLI*.

UEFI Support

Titanium Server supports hosts that use UEFI.

The Unified Extensible Firmware Interface (UEFI) specification replaces the traditional hardware-specific BIOS with a standards-based extensible interface between the hardware platform and the operating system. This overcomes the design limitations of BIOS for greatly improved hardware support. Titanium Server supports PXE boot for hosts configured to use either BIOS or UEFI.

OAM Network

OAM Network Firewall

Titanium Server supports the ability to override or augment the default **iptables** rules for the built-in OAM network firewall.

The option to use a custom firewall is available at system installation. For more information, see *Titanium Server Installation*.

Alarm Generation and Reporting

Titanium Server provides facilities for alarm and log management.

Support is included for the following:

- alarms on performance management thresholds and cloud-level services and equipment
- customer logs
- SNMPv2ct

As part of alarm management, Titanium Server also provides support for alarm suppression.

For more information, see *Titanium Server System Administration: Alarm Messages*.

REST API Support

Titanium Server supports external REST APIs through its OAM floating IP address.

The following REST APIs are supported:

- OpenStack REST APIS

Titanium Server supports selected OpenStack REST API functionality, including:

- Block Storage API (with Titanium Server extensions)
- Compute API (with Titanium Server extensions)
- Identity API
- Image Service API
- Networking API (with Titanium Server extensions)
- Orchestration API
- Telemetry API (with Titanium Server extensions)

For a complete list of supported and unsupported OpenStack REST APIs, refer to the REST API documentation included with the Titanium Server Software Development Kit.

- Titanium Server REST APIs
 - SysInv API
 - Patching API



NOTE: Titanium Server extensions use the OpenStack Extension Mechanism to ensure compatibility with existing clients.

HTTP or HTTPS protocol can be supported for these external REST APIs.

Web Administration Interface Improvements

Titanium Server web administration interface is an enhanced version of the Horizon web interface provided by OpenStack.

The enhancements include:

- Automatic page refresh for immediate notification of status changes. This eliminates the need for manual reloading of web pages to access up-to-date information about the cloud
- Admin Overview page with charts and tables providing a high-level overview of cloud resources
 - avg/max/min compute vCPU usage
 - avg/max/min compute memory usage
 - avg/max/min compute disk usage
 - avg/max/min AVS CPU utilization
 - avg/max/min provider network port utilization
 - current host status (available, unavailable, locked, unlocked)
- Improved performance management resource usage web page
 - optimized performance management sample DB queries for improved usability
 - capability to filter on fields in the performance management sample's metadata
 - human-readable legends and chart labels using object names instead of long UUID text strings
 - performance management meters can be selected using their brief descriptions
 - meters for CPU, memory, and disk utilization
- Ability to re-brand the Horizon GUI by modifying color schemes, logos, icons, and server identification.

Centralized Log Collection and Analysis

You can configure Titanium Server to send detailed system logs from all hosts to a remote log server for centralized review and analysis.

You can also set up your own dashboards for viewing log details.

When centralized logging is enabled, logs written to the **/var/log** directory on each host are also sent to a remote log server. System logs used for troubleshooting and advanced analysis are included, along with the customer logs normally accessed from the CLI or web administration interface. By centralizing the logs, advanced users can apply powerful searches and advanced visualizations to examine the behavior of the system.

Each host sends logs through the Active Controller to a remote log server over the OAM network using either TCP or UDP. For added security, you can optionally configure a TLS connection. At the log server, an ELK (Elasticsearch, Kibana, Logstash) stack collects and presents the information.

Logstash

collects the logs

Elasticsearch

provides a search engine

Kibana

provides web-based visualization

The ELK stack is a widely-used source log analytics engine, with ample documentation and tutorials on the Internet. The Titanium Server SDK includes custom searches and filters for use with a Titanium Server system.

To set up and use a log server, see *Titanium Server System Administration: Configuring Centralized Log Collection*.

Performance Management Extensions

The Ceilometer service included with Titanium Server features improved performance, scalability, and usability.

It has been extended to support a CSV-formatted file backend that provides a more traditional Telco Northbound interface for performance management. Titanium Server also adds performance monitoring for platform resource usage (CPU, memory, and disk), and for the vSwitch process.

Overview of Remote CLI SDK Module

The Titanium Server remote CLI software package provides a set of CLI commands that can be installed on a remote workstation and used to manage Titanium Server remotely.

This package is delivered in Titanium Server's **wrs-remote-clients** SDK Module and is installable on any Ubuntu or CentOS host. The host must have network connectivity to the OAM IP address of the Titanium Server system in order to execute remote CLI commands against that Titanium Server system. The Titanium Server remote CLI module supports both an admin role user and a non-admin role user. It also provides support for both HTTP and HTTPS access, and Keystone authentication. This enables access to the Titanium Server CLI without requiring the user to have SSH login access to Titanium Server controllers.



NOTE: For HTTPS access, if Titanium Server uses the self-signed digital certificate included for demonstration purposes, then CLI commands from the remote client must be used in insecure mode (typically by including an **--insecure** option) to accept the certificate without verifying it.

CLI clients supported by Titanium Server's remote CLI access include:

- Nova
- Neutron
- Cinder
- Glance
- Keystone
- Ceilometer
- Heat
- System
- OpenStack (the new OpenStack all-in-one CLI Client)

The remote CLI module does not support the Titanium Server patching CLI Client.

In addition to being able to send CLI commands to a Titanium Server OpenStack system or region, the Remote Titanium Server CLI Client can also be used to send CLI commands to open-source OpenStack and/or non-Titanium Server OpenStack systems or regions.



NOTE: The Titanium Server remote CLI client and the open-source OpenStack remote CLI client cannot be installed on the same host; only one CLI client or the other can be installed on a host.

Authentication

All CLI commands within the Titanium Server remote CLI client are authenticated through Keystone. Authentication can be specified using the CLI, or retrieved from a shell environment. For convenience, you can download an RC file from the Titanium Server web administration interface and use it to export authentication parameters into the Linux shell. For more information, see *Using an Open RC File for Remote CLI Access*.

Software Management

Patching Facilities

Titanium Server includes tools to patch system images to ensure they are always up to date with the latest release and security fixes.

For more information, see *Titanium Server Patching and Upgrading Platform Software: Managing Software Patches*.

Backup and Restore Facilities

Titanium Server includes tools to backup and restore system data, virtual machines, and storage resources.

For more information, see *Titanium Server Patching and Upgrading Platform Software: System Data Backup with Controller Storage and Performing a System Restore*.

Upgrade Capabilities

Titanium Server provides comprehensive in-service upgrade capabilities.

Software upgrades move Titanium Server software from one release to the next and change the version of updated software components. The upgrade typically updates components that may include the kernel, Operating System packages, OpenStack, and Titanium Server specific software. The software upgrade process manages several complexities such as conversion of database schemas, conversion of database data, API compatibility management between Titanium Server Hosts, and live migrating of hosted VMs.

Patch Orchestration Capabilities

Titanium Server supports patch orchestration, which allows an entire Titanium Server system to be patched with a single operation.

Patch orchestration can be configured and run through the CLI, the Horizon GUI or the VIM REST API.

Patch orchestration will automatically iterate through all nodes of the system and install the applied patch(es) to each node; first the Controller Nodes, then the Storage Nodes and finally the Compute Nodes. During the patching of Compute Nodes, the migration of VMs off of Compute Nodes being patched, is managed automatically by Patch Orchestration. The Controller Nodes are always patched in serial, however the Storage Nodes and Compute Nodes can configurably be patched in parallel in order to reduce the overall time of installing the patch.

Patch orchestration can install 1 or more applied patches at the same time, and can install Reboot-Required Patches and/or In-Service Patches at the same time. Patch orchestration will only lock/unlock (that is, reboot) a node to install a patch if at least 1 Reboot-Required patch has been applied.

Integrated OpenStack Services

Titanium Server integrates support for OpenStack services.

The following OpenStack services are integrated into Titanium Server:

- Nova
- Neutron
- Keystone
- Glance
- Cinder
- Horizon
- Ceilometer
- Heat
- Swift

For more information, see [Supported Openstack Features](#) on page 25.

Ceph-backed VM Storage

Titanium Server provides a flexible and scalable range of options for Ceph-backed storage.

Systems configured for Ceph-backed storage use dedicated storage hosts to provide storage resources for VMs. You can use these resources for VM swap, ephemeral, and boot-from-image root disks (remote ephemeral storage) to support persistence for live migration, cold migration, and evacuation. You can optionally enable additional Ceph resources to support Openstack object storage (Swift) access through the Ceph Object Gateway, so that Swift objects are accessible to VMs for use during operation.

Titanium Server supports up to four pairs of storage hosts for scalability. Allocations for ephemeral, image, object, and volume storage are rebalanced dynamically to meet changing requirements.

Systems configured for Ceph-backed dedicated storage can use cache tiering to improve performance for frequently-accessed data. For more information, see *Titanium Server System Administration: Cache Tiering*.

Systems configured for controller-based storage can be extended to use Ceph storage. For more information, see *Titanium Server System Administration: Dedicated Storage for Systems Using Controller Storage*.

Heat

Heat Extensions

The OpenStack Heat orchestration service is enhanced.

Enhancements include:

- Support for several Titanium Server-specific resource types (for example,. ProviderNet, ProviderNetRange, QoSPolicy, a new scaling policy)
- Simplified naming options for flavors and VM instances
- Enhanced server group support, including best-effort instantiation and group size limiting
- Simplified options for passing user data to an instance
- Improved stack access for users
- Support for dependencies during resource allocation
- Local-file or URL locations for Glance images
- Helpful example templates illustrating real reference scenarios

Resource Scaling (Autoscaling)

You can use Heat to reassign stack resources automatically to meet changing conditions.

You can define and monitor performance thresholds for metrics such as CPU activity, and then add or remove resources when the thresholds are crossed. This allows you to make efficient use of the hardware in the cluster, by allocating resources only when they are needed, and assigning them where they are most required.

Titanium Server supports two types of scaling:

In/Out

This type of scaling (also known as *horizontal scaling*) adds or removes instances as needed.

Up/Down

This type of scaling (also known as *vertical scaling*) increases or decreases resources (for example, vCPUs) for individual instances as needed. For more about up/down scaling, see *Titanium Server Tenant User's Guide: Scaling Virtual Machine Resources*.

Performance metrics can be collected and reported by the Titanium Server platform, or by the guests using guest metrics.

Titanium Server SDK

The Titanium Server Software Development Kit (SDK) provides drivers, daemons, API libraries, and configuration files that you can include in a guest image to leverage the extended capabilities of Titanium Server.

Available components include:

- Accelerated Kernel Network Drivers—Drivers for improved performance of kernel-based networking VNFs
- Accelerated DPDK Network Drivers—Drivers for high-performance DPDK-based networking VNFs
- VM Resource Scaling—A service for scaling the capacity of a guest server on demand
- Guest Heartbeat—A service for health monitoring of guest applications
- Server Group Messaging—A service for low-bandwidth peer messaging between servers in the same group
- SNMP MIB—Resources for system alarms management
- Sample Heat Orchestration Templates—Resources for deploying and managing stacks of applications or application services
- OpenStack REST API Documentation—Documentation for Titanium Server REST APIs and Titanium Server extensions to OpenStack REST APIs
- Configuration utilities—Tools for generating and validating files used to automate Titanium Server installation and configuration

- ELK Log Server—A utility for installing and configuring a remote log server to collect logs from all Titanium Server hosts for search and analysis using an ELK stack
- Custom Branding—Resources for customizing the Horizon GUI

For more information, see the *Titanium Server Software Development Kit*.

Supported OpenStack Features

Supported Openstack Features 25

Supported Openstack Features

This section documents the supported and unsupported items within the upstream opensource OpenStack REST APIs.

Supported indicates that the functionality works and has been explicitly verified.

Unsupported indicates that the functionality either will not work or has not been explicitly verified.

Compute API v2.1

Supported REST APIs	
API versions	/
Limits	/v2.1/{tenant_id}/limits
Extensions	/v2.1/{tenant_id}/extensions[/*]
Servers	/v2.1/{tenant_id}/servers
	/v2.1/{tenant_id}/servers/detail
	/v2.1/{tenant_id}/servers/{server_id}
Server metadata	/v2.1/{tenant_id}/servers/metadata[/*]
Server addresses	/v2.1/{tenant_id}/servers/ips
Server actions	/v2.1/{tenant_id}/servers/action

Supported REST APIs	
Flavors	/v2.1/flavors
	/v2.1/flavors/detail
	/v2.1/flavors/{flavor_id}
Images	/v2.1/images
	/v2.1/images/detail
	/v2.1/images/{image_id}
Image metadata	/v2.1/images/{image_id}/metadata[/*]
Servers with block device mapping format	/v2.1/{tenant_id}/servers
Servers with configuration drive	/v2.1/{tenant_id}/servers
	/v2.1/{tenant_id}/servers/{server_id}
	/v2.1/{tenant_id}/servers/{server_id}/detail
Servers console	/v2.1/{tenant_id}/servers/{server_id}/action
Servers console output	/v2.1/{tenant_id}/servers/{server_id}/action
Servers extended attributes	/v2.1/{tenant_id}/servers
	/v2.1/{tenant_id}/servers/{server_id}
Servers with extended availability zones	/v2.1/{tenant_id}/servers/{server_id}
	/v2.1/{tenant_id}/servers/detail
Servers extended status	/v2.1/{tenant_id}/servers/{server_id}
	/v2.1/{tenant_id}/servers/detail
Servers with IP type	/v2.1/{tenant_id}/servers/{server_id}/action
	/v2.1/{tenant_id}/servers/detail
Servers multiple create	/v2.1/{tenant_id}/servers
Servers with scheduler hints	/v2.1/{tenant_id}/servers
Servers administrative actions	/v2.1/{tenant_id}/servers/{server_id}/action
	Exceptions: (i.e. NOT Supported)
	Reset networking on server
	Inject network information
	Create server backup
Servers deferred delete	/v2.1/{tenant_id}/servers/{server_id}/action

Supported REST APIs	
Servers start and stop	/v2.1/{tenant_id}/servers/{server_id}/action
Servers and images with disk config	/v2.1/{tenant_id}/servers/*]
	/v2.1/{tenant_id}/images/*]
Servers availability zones	/v2.1/{tenant_id}/servers
	/v2.1/os-availability-zone[/*]
Servers virtual interfaces	/v2.1/{tenant_id}/servers/{server_id}/os-virtual-interfaces
Servers with volume attachments	/v2.1/{tenant_id}/servers/{server_id}/os-volume_attachments[/*]
Server boot from volume	/v2.1/{tenant_id}/os-volumes_boot
Flavors create and delete	/v2.1/{tenant_id}/flavors[/{flavor_id}]
Flavors with extra-specs	/v2.1/{tenant_id}/flavors/{flavor_id}/os-extra_specs[/{key_id}]
Images with size attribute	/v2.1/{tenant_id}/images/detail
	/v2.1/{tenant_id}/images/{image_id}
Limits with project usage	/v2.1/{tenant_id}/limits
Limits with project usage for administrators	/v2.1/{tenant_id}/limits/{tenant_id}
Host aggregates	/v2.1/{tenant_id}/os-aggregates[/{aggregate_id}]
	/v2.1/{tenant_id}/os-aggregates/{aggregate_id}/action
Fixed IPs	/v2.1/{tenant_id}/os-fixed-ips/{fixed_ip}[/action]
Fixed IP DNS records	/v2.1/{tenant_id}/os-floating-ip-dns[/*]
Floating IP pools	/v2.1/{tenant_id}/os-floating-ip-pools
Floating IPs	/v2.1/{tenant_id}/os-floating-ips[/{id}]
	/v2.1/{tenant_id}/servers/{server_id}/action
Floating IPs bulk	/v2.1/{tenant_id}/os-floating-ips-bulk[/*]
Hypervisors	/v2.1/{tenant_id}/os-hypervisors[/*]
Hypervisor with status	/v2.1/{tenant_id}/os-hypervisor-status/detail
Key Pairs	/v2.1/{tenant_id}/os-key pairs[/{key pair_name}]

Supported REST APIs	
Migrations	/v2.1/{tenant_id}/os-migrations
Networks	/v2.1/{tenant_id}/os-networks[/*]
Quota class	/v2.1/os-quota-class-sets/{class_id}
Quota sets	/v2.1/{tenant_id}/os-quota-sets/{tenant_id}[/*]
Security groups	/v2.1/{tenant_id}/os-security-groups[/*]
Rules for security group	/v2.1/{tenant_id}/os-security-group-rules[/*]
Rules for default security group	/v2.1/{tenant_id}/os-security-group-default-rules[/*]
Server Groups	/v2.1/{tenant_id}/os-server-groups[/*]
Volume extension	/v2.1/{tenant_id}/os-volumes[/*]
	/v2.1/{tenant_id}/os-snapshots[/*]

Networking API v2.0

Supported REST APIs	
Networking	/v2.0/networks[/*]
	/v2.0/subnets[/*]
	/v2.0/ports[/*]

Networking API v2.0 extensions

Supported REST APIs	
Extensions	/v2.0/extensions[/*]
Quotas extension	/v2.0/quotas[/*]
Networks provider extended attributes	/v2.0/networks[/*]
Networks multiple provider extension	/v2.0/networks[/*]
VLAN transparency extension	/v2.0/networks[/*]
Ports binding extended attributes	/v2.0/ports[/*]
Security groups and rules	/v2.0/security-groups[/*]
	/v2.0/security-group-rules[/*]
Layer-3 newtorking	/v2.0/routers[/*]

Supported REST APIs	
	EXCEPTION: (not supported) /v2.0/routers: ha attribute
	NOTE: Titanium Server Router Instances are HA protected across compute nodes.
	/v2.0/floatingips[/*]
Extra routes	/v2.0/networks[/*]
Unsupported REST APIs	
Metering labels and rules	/v2.0/metering[/*]
Load-Balancer-as-a-Service	/v2.0/lb[/*]
	/v2.0/lbaas[/*]
Virtual-Private-Network-as-a-Service	/v2.0/vpn[/*]

Block Storage API v2.0

Supported REST APIs	
API Versions	/
API extensions	/v2/{tenant_id}/extensions[/*]
Volumes	/v2/{tenant_id}/volumes[{volume_id}]
	/v2/{tenant_id}/volumes/detail
Volume actions	/v2/{tenant_id}/volumes/{volume_id}/action
Volume types	/v2/{tenant_id}/types[{volume_type_id}]
Snapshots	/v2/{tenant_id}/snapshots[{snapshot_id}[/metadata]]
	/v2/{tenant_id}/snapshots/detail
Quota sets extension	/v2/{tenant_id}/os-quota-sets[/*]
Limits extension	/v2/{tenant_id}/limits
Backups	/v2/{tenant_id}/backups[{backup_id}[/restore]]
	/v2/{tenant_id}/backups/detail
Backup actions	/v2/{tenant_id}/backups/{backup_id}/action

Supported REST APIs	
Volume image metadata extension	/v2/{tenant_id}/os-vol-image-meta
Volume type access	/v2/{tenant_id}/volumes
Unsupported REST APIs	
Quality of service (QoS) specifications	/v2/{tenant_id}/qos-specs[/[*]]
Volume manage extension	/v2/{tenant_id}/os-volume-manage

Identity API v2.0

This API is not supported.

Identity Admin API v2.0

This API is not supported.

Identity API v2.0 extensions

This API is not supported.

Identity API v3.0

Supported REST APIs	
Authentication and token mgmt	/v3/auth/tokens
	/v3/auth/catalog
	/v3/auth/projects
	/v3/auth/domains
Credentials	/v3/credentials[/ {credential_id}]
Domains	/v3/domains[/ {domain_id}]
Groups	/v3/groups[/ {group_id}]
	/v3/groups/{group_id}/users[/ {user_id}]
Policies	/v3/policies[/ {policy_id}]
Projects	/v3/projects[/ {project_id}]
Regions	/v3/regions[/ {region_id}]
Roles	/v3/roles[/ {role_id}]

Supported REST APIs	
	/v3/domains/{domain_id}/groups/{group_id}/roles[{role_id}]
	/v3/projects/{project_id}/groups/{group_id}/roles[{role_id}]
	/v3/roles/{prior_role_id}/implies[{implies_role_id}]
	/v3/role_assignments
	/v3/role_inferences
Service catalog and endpoints	/v3/services[{service_id}]
	/v3/endpoints[{endpoint_id}]
Users	/v3/users[{user_id}]
	/v3/users/{users_id}/groups
	/v3/users/{users_id}/projects
	/v3/users/{users_id}/password
Unsupported REST APIs	
Domain configuration	/v3/domains/config/default
	/v3/domains/config/{group}/default
	/v3/domains/config/{group}/{option}/default
	/v3/domains/{domain_id}/config[{group}][/{option}]]
OS-INHERIT API	/v3/OS-INHERIT/domains/{domain_id}/users/{user_id}/roles[{role_id}]/inherited_to_projects
	/v3/role_assignments

Identity API v3.0 extensions

This API is NOT Supported.

Image Service API v1.0

Supported REST APIs	
API Versions	/
Images	/v1/images[{image_id}]

Supported REST APIs	
	/v1/images/detail
Unsupported REST APIs	
Members	/v1/images/{image_id}/members[/ {owner_id}]
Shared images	/v1/images/{owner}

Object Storage API v1.0

Supported REST APIs	/info
Discoverability	/info
Accounts	/v1/{account} NOTE: no support for custom metadata; only supports a subset of Swift ACLs.
Containers	/v1/{account}[/ {container}]
Objects	/v1/{account}[/ {container}[/ {object}]] NOTE: no support for expiring objects, no support for object versioning, no support for CORS, no support for Static Website
Endpoints	/v1/endpoints

Orchestration API v1.0

Supported REST APIs	
API versions	/
Stacks	/v1/{tenant_id}/stacks[/ {stack_name}]
	/v1/{tenant_id}/stacks/{stack_name}/ resources
	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}
	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/snapshots[/ *]
	EXCEPTION: (not supported)
	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/abandon

Supported REST APIs	
Stack actions	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/actions
Stack resources	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/resources[/*]
Stack events	/v1/{tenant_id}/stacks/{stack_name}[/ {stack_id}]/events[/*]
	EXCEPTION: (not supported)
	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/resources/{resource_name}/ events[/{event_id}]
Manage service	/v1/{tenant_id}/services
Templates	/v1/{tenant_id}/stacks/{stack_name}/ {stack_id}/template
	/vi/{tenant_id}/validate
	EXCEPTION: (not supported)
	/vi/{tenant_id}/template_versions
	/vi/{tenant_id}/resource_types/{type_name}/ template
Software configuration	/v1/{tenant_id}/software_configs[/*]
	/v1/{tenant_id}/software_deployments[/*]
Unsupported REST APIs	
Build info	/v1/{tenant_id}/build_info

Telemetry API v2.0

Supported REST APIs	
Alarms	/v2/alarms[/*]
Meters	/v2/meters[/*]
Samples	/v2/samples[/*]
Resources	/v2/resources[/*]
Capabilities	/v2/capabilities

4

Virtual Applications

Virtual Applications 35

Virtual Applications

Titanium Server provides a flexible feature-rich cloud platform for hosting a wide variety of Virtual Applications ranging from Telecom to Industrial, Central Office to Customer Premise and much more.

Titanium Server's strength and flexibility in providing a Highly Available hosting environment allows it to host Virtual Applications in Telecom scenarios and Industrial scenarios that demand such high availability and robustness. A variety of HA application models can be hosted ranging from classic cloud protection pools to traditional Telco or Industrial 1:1 or 1:N Hot Standby Redundancy.

Titanium Servers scalable and high performance Networking solution enables Telcos to include not only their Management and Control / Signaling Applications in their cloud-based solution, but also their high bandwidth Data Plane Applications as well. Titanium Server provides DPDK-accelerated L2 and L3 platform services and enables network-performance-optimized hosting of DPDK-based Telco Data Plane Applications.

Titanium Server also provides an ideal hosting environment for Virtual Applications demanding high compute performance and/or high database capacity and performance, such as Home Subscriber Server (HSS) / Home Location Register (HLR) applications in Telco Central Offices. Titanium Server provides fine-tuned control over virtual CPU policies and scheduling as well as optimizes compute performance thru huge page memory performances. For storage capacity and performance, Titanium Server provides support for a highly scalable CEPH storage cluster with SSD-based journaling or SSD-based full cache tiering for economical storage performance at large scale.

Titanium Server's flexibility in deployment options ranges from hundreds of nodes down to just a single pair of servers. This allows Titanium Server to be deployed in central large-scale Telecom Data Centers or on customer premises in vCPE type solutions.

Titanium Server can provide hosting for the complete range of Telecom Applications: Management Plane Applications, Control / Signaling Plane Applications and Data Plane Applications. This enables end-users of Titanium Server to deploy complete end-to-end Telco solutions on Titanium Server; examples of which include virtual Evolved Packet Core (vEPC), virtual IP Multimedia Subsystem (vIMS) and the new 5G networks. To accelerate the deployment of all these use cases, Telcos can leverage the extensive range of products from industry-leading partners that have been validated through Wind River's Titanium Cloud ecosystem.