

# Prism - Prism Architecture

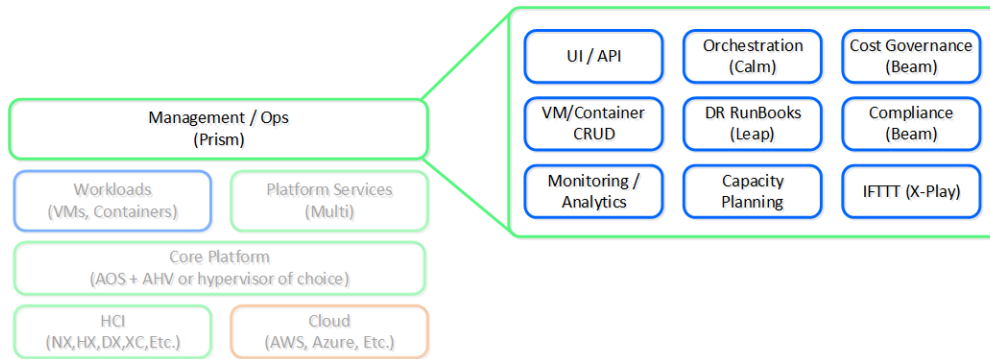
[ PDF generated October 31 2024. For all recent updates please see the Nutanix Bible releases notes located at [https://nutanixbible.com/release\\_notes.html](https://nutanixbible.com/release_notes.html). Disclaimer: Downloaded PDFs may not always contain the latest information. ]

Prism is a distributed resource management platform which allows users to manage and monitor objects and services across their Nutanix environment, whether hosted locally or in the cloud.

These capabilities are broken down into two key categories:

- Interfaces
  - HTML5 UI, REST API, CLI, PowerShell CMDlets, etc.
- Management Capabilities
  - Platform management, VM / Container CRUD, policy definition and compliance, service design and status, analytics and monitoring

The following figure illustrates the conceptual nature of Prism as part of the Nutanix platform:

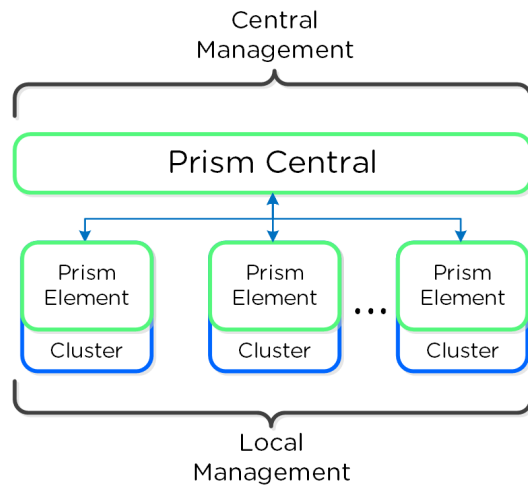


## High-Level Prism Architecture

Prism is broken down into two main components:

- Prism Central (PC)
  - Multi-cluster manager responsible for managing multiple Nutanix Clusters to provide a single, centralized management interface. Prism Central is an optional software appliance (VM) which can be deployed in addition to the AOS Cluster (can run on it).
  - 1-to-many cluster manager
- Prism Element (PE)
  - Localized cluster manager responsible for local cluster management and operations. Every Nutanix Cluster has Prism Element built-in.
  - 1-to-1 cluster manager

The figure shows an image illustrating the conceptual relationship between Prism Central and Prism Element:



Prism Architecture

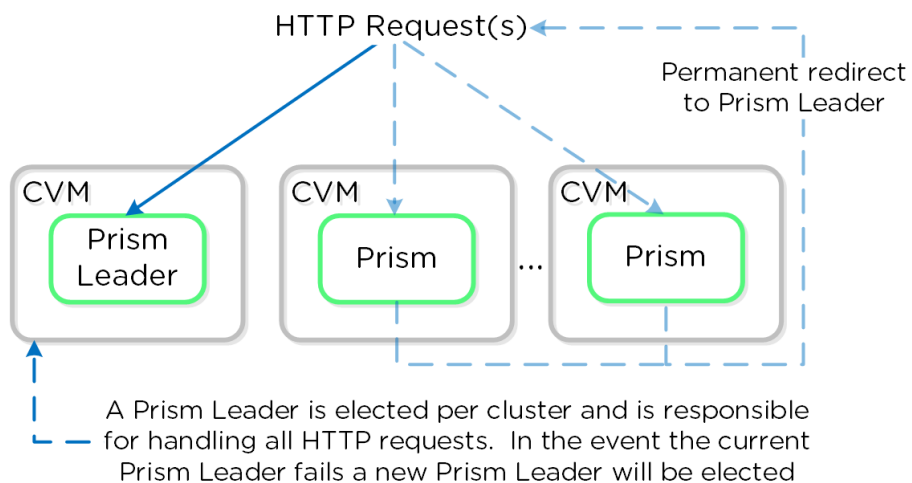
**Pro tip**

For larger or distributed deployments (e.g. more than one cluster or multiple sites) it is recommended to use Prism Central to simplify operations and provide a single management UI for all clusters / sites.

**Prism Services**

A Prism service runs on every CVM with an elected Prism Leader which is responsible for handling HTTP requests. Similar to other components which have a Leader, if the Prism Leader fails, a new one will be elected. When a CVM which is not the Prism Leader gets a HTTP request it will permanently redirect the request to the current Prism Leader using HTTP response status code 301.

Here we show a conceptual view of the Prism services and how HTTP request(s) are handled:



Prism Services - Request Handling

**Prism ports**

Prism listens on ports 80 and 9440, if HTTP traffic comes in on port 80 it is redirected to HTTPS on port 9440.

When using the cluster external IP (recommended), it will always be hosted by the current Prism Leader. In the event of a Prism Leader failure the cluster IP will be assumed by the newly elected Prism Leader and a gratuitous ARP (gARP) will be used to clean any stale ARP cache entries. In this scenario any time the cluster IP is used to access Prism, no redirection is necessary as that will already be the Prism Leader.

### Pro tip

You can determine the current Prism leader by running the following command on any CVM:

```
curl localhost:2019/prism/leader
```

## Authentication and Access Control (RBAC)

### Authentication

Prism currently supports integrations with the following authentication providers:

- Prism Element (PE)
  - Local
  - Active Directory
  - LDAP
- Prism Central (PC)
  - Local
  - Active Directory
  - LDAP
  - SAML Authn (IDP)

### SAML / 2FA

SAML Authn allows Prism to integrate with external identity providers (IDP) that are SAML compliant (e.g. Okta, ADFS, etc.).

This also allows you to leverage the multi-factor authentication (MFA) / two-factor authentication (2FA) capabilities these providers support for users logging into Prism.

### Access Control

Coming soon!