Driving organisational adoption of DevOps, Cloud and Microservices

Official Supplier to HM Government for three consecutive years
G-Cloud 9
G-Cloud 10
G-Cloud 11

HM Government
G-Cloud
Supplier

G-Cloud 11 - Hashicorp Enterprise Software Licenses

www.nuaware.com 0203 488 0530 info@nuaware.com

Hashicorp Product Suite

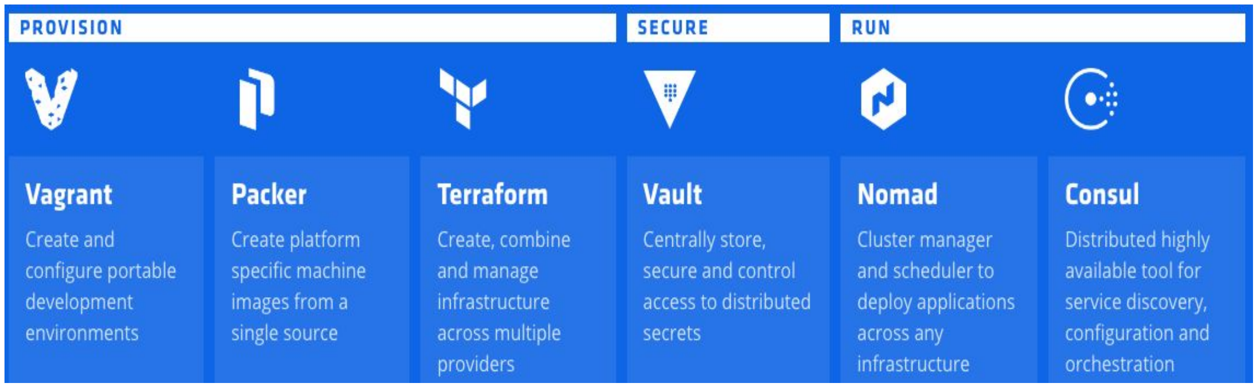
Hashicorp offer an ecosystem of tools with the goal of revolutionising datacenter management: application development, delivery, and maintenance. HashiCorp makes Vagrant, a tool for building complete development environments. With an easy-to-use workflow and focus on automation, Vagrant lowers development environment setup time, increases development/production parity, and makes the "works on my machine" excuse a relic of the past. Since then the team have gone on to offer disparate but highly integratable solutions based around operating heterogeneous and highly complex infrastructure, at scale.

With tools like Consul, Nomad, Vault, Terraform, Packer and Vagrant, Hashicorp offers a complete stack of tools to ensure that you can minimizing the challenges of shipping, rapidly iterating, and securing software applications.

HashiCorp delivers pragmatic solutions to maximize developer and operator agility. HashiCorp tools build microservice-driven applications that are codified, automated, scalable, and secure. Nuaware is a HashiCorp premium system integration partner. We provide professional services for deployment, architecture, training, and custom development.

Hashicorp Tools

All of HashiCorp's foundational technologies are open source. These tools are designed to address the realities of datacenter heterogeneity: physical machines, virtual machines, containers, serverless architectures and whatever comes after that. The focus is on workflows, not technologies.

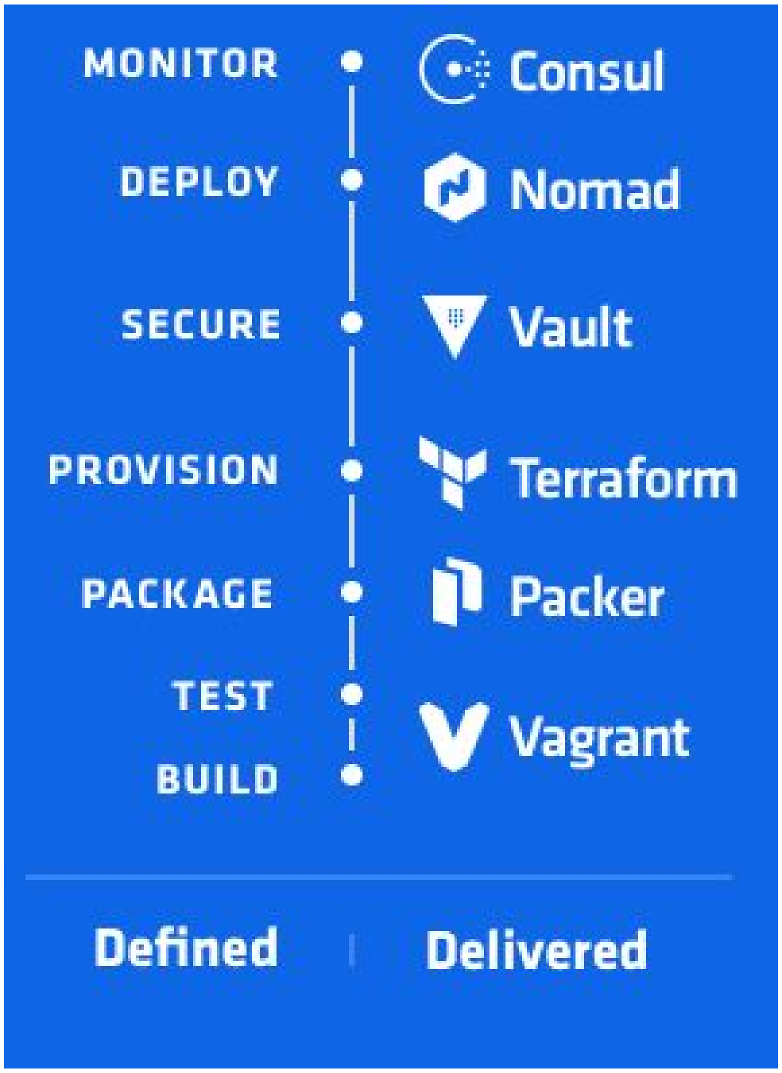


PROVISION			SECURE	RUN	
Vagrant	**Packer**	**Terraform**	**Vault**	**Nomad**	**Consul**
Create and configure portable development environments	Create platform specific machine images from a single source	Create, combine and manage infrastructure across multiple providers	Centrally store, secure and control access to distributed secrets	Cluster manager and scheduler to deploy applications across any infrastructure	Distributed highly available tool for service discovery, configuration and orchestration

www.nuaware.com **0203 488 0530** info@nuaware.com

The definition of DevOps varies from business to business, but the zeitgeist of DevOps is about minimizing the challenges of shipping, rapidly iterating, and securing software applications. HashiCorp defines DevOps as an organizational process tied to the needs of modern applications, with a focus on empowering individuals to improve agility.

DevOps is about minimizing the challenges of shipping, rapidly iterating, and securing software applications.



DevOps Defined

Provision, Secure and Run Any Infrastructure For Any Application

Every organization has slightly different elements in its software delivery process, driven by technology choices, compliance requirements, or other factors. But if you look at the whole forest and not just the trees, there are seven elements to the software delivery lifecycle:

BUILD An application starts with a developer writing code. For a new application the initial version must be written, but for existing applications there is a perpetual cycle of adding new features and functionality, fixing bugs, and improving performance. This element largely involves developers, but operations teams may be responsible for providing the environment and tools developers are using to write code.

TEST While an application is being written and prior to release, it goes through multiple types of testing. The simplest tests, unit testing, is done by developers, and can be layered with integration testing, acceptance testing, end-to-end tests, etc. This is an important part of the application delivery lifecycle, as it provides automated feedback and an important risk management control. This largely involves developers, but also dedicated QA teams and operations teams who may own the testing infrastructure.

PACKAGE Once an application is written and tested, it needs to be packaged for production. The packaging can depend heavily on the technology or target environment. For example, this might be a WAR file for JBoss, or a Docker container for Kubernetes. The underlying application is being packaged from its raw source form into a packaged executable for the target environment. These packages are stored in a registry such as Artifactory.

PROVISION Applications require somewhere to run. Under all the layers of abstraction there is still compute, storage, and networking resources that must be provided. These might be provided directly with bare metal or VMs, or indirectly through a PaaS or Lambda-as-a-Service frameworks. In any case, these resources must be provisioned and configured to the application requirements, updated over time, and finally decommissioned at the end of their utility. Provisioning is usually owned by operations teams and provided to developers.

SECURE The overall security of a system is only as strong as its weakest link. This means security is involved from the beginning of the application delivery process. Security teams help ensure best practices are used during development, assist in modeling network topology, protect credentials used to provision infrastructure, and grant secrets needed to deploy applications such as database passwords and API tokens. Security typically falls to dedicated roles on a teams, but involves all other teams that are delivering an application.

DEPLOY With the underlying resources provisioned, deployment means taking the packaged application and running it. This can be tightly coupled with provisioning if machines or VMs are specialized to run a single application, or decoupled if a scheduler is used to dynamically place applications on machines.

MONITOR Running applications need to be monitored to ensure they continue to run and are healthy. Services need to talk to each other while avoiding communicating with faulty or degraded instances. This leads to the need for service discovery. Monitoring ranges a wide gamut from coarse grained liveness to detailed logging and telemetry. Monitoring involves developers who want to understand the behavior of their applications, operators who manage the infrastructure, and monitoring and site reliability teams who maintain the broader application.

DevOps Delivered

Provision, Secure and Run Any Infrastructure For Any Application

Designing a high performance organization is similar to designing a high performance application: it must require minimal coordination. For software this is best captured by Amdahl's law, but if we think of individuals as "serial execution units" productivity is similarly dominated by coordination.

The seven elements of delivering an application cannot be skipped, so the priority must become minimizing the coordination required to perform each step. If each team is empowered to work independently, then coordination can be reduced and individual productivity can be increased; application delivery velocity increases. This is the heart of DevOps, and the tools we choose must prioritize these key functions. To maintain a consistent process, those tools must focus on workflows, not technologies, to address the technical heterogeneity that is the reality of most organization.
HashiCorp provides a suite of tools with DevOps in-mind focused on reducing manual coordination across the elements of application delivery lifecycle.

Vagrant

Development Environments Made Easy

Vagrant allows developers to quickly setup a development environment on their own, without needing to consult peers or operators. By providing a production-like environment, it also allows developers to easily test their code with a tighter feedback loop. This is one of the goals of Continuous Integration (CI), as it allows developers to be more individually productive by giving them feedback more rapidly.

Simple and Powerful

Vagrant provides the same, easy workflow regardless of your role as a developer, operator, or designer. It leverages a declarative configuration file which describes all your software requirements, packages, operating system configuration, users, and more.

Production Parity

The cost of fixing a bug exponentially increases the closer it gets to production. Vagrant aims to mirror production environments by providing the same operating system, packages, users, and configurations, all while giving users the flexibility to use their favorite editor, IDE, and browser. Vagrant also integrates with your existing configuration management tooling like Chef, Puppet, Ansible, or Salt, so you can use the same scripts to configure Vagrant as production.

Works where you work

Vagrant works on Mac, Linux, Windows, and more. Remote development environments force users to give up their favorite editors and programs. Vagrant works on your local system with the tools you're already familiar with. Easily code in your favorite text editor, edit images in your favorite manipulation program, and debug using your favorite tools, all from the comfort of your local laptop.

Packer

Build Automated Machine Images

Packer provides a single workflow to package applications for any target environment. By sharing configuration, Packer allows teams to be decoupled and avoid coordination. The coordination is pushed into the artifact registry such as Artifactory or Docker Hub.

Modern, Automated

Packer is easy to use and automates the creation of any type of machine image. It embraces modern configuration management by encouraging you to use automated scripts to install and configure the software within your Packer-made images. Packer brings machine images into the modern age, unlocking untapped potential and opening new opportunities.

Works Out of The Box

Out of the box Packer comes with support to build images for Amazon EC2, CloudStack, DigitalOcean, Docker, Google Compute Engine, Microsoft Azure, QEMU, VirtualBox, VMware, and more. Support for more platforms is on the way, and anyone can add new platforms via plugins.

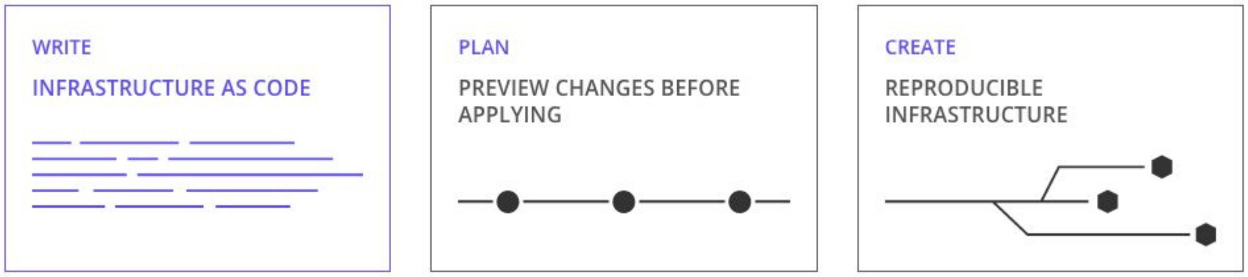
Terraform

Write, Plan, and Create Infrastructure as Code

Terraform is a product to provision infrastructure and application resources across any infrastructure using a common workflow. Terraform enables operators to safely and predictably create, change, and improve production infrastructure. It codifies APIs into declarative configuration files that can be shared amongst team members, treated as code, edited, reviewed, and versioned.

<u>Simple and Powerful</u>

Terraform enables you to safely and predictably create, change, and improve production infrastructure. It is an open source tool that codifies APIs into declarative configuration files that can be shared amongst team members, treated as code, edited, reviewed, and versioned.



INFRASTRUCTURE AS CODE

Define infrastructure as code to increase operator productivity and transparency.

<u>COLLABORATE & SHARE</u>

Terraform configuration can be stored in version control, shared, and collaborated on by teams of operators.

<u>EVOLVE YOUR INFRASTRUCTURE</u>

Track the complete history of infrastructure versions.

<u>AUTOMATION FRIENDLY</u>

If it can be codified, it can be automated.

nuaware

ONE SAFE WORKFLOW ACROSS PROVIDERS

Terraform provides an elegant user experience for operators to safely and predictably make changes to infrastructure.

MAP RESOURCE DEPENDENCIES

Understand how a minor change could have potential cascading effects across an infrastructure before executing that change. Terraform builds a dependency graph from the configurations, and walks this graph to generate plans, refresh state, and more.

SEPARATION OF PLAN & APPLY

Separating plans and applies reduces mistakes and uncertainty at scale. Plans show operators what would happen, applies execute changes.

ONE SAFE WORKFLOW

Use Terraform to create resources across all major infrastructure providers (AWS, GCP, Azure, OpenStack, VMware, and more).

REPRODUCIBLE INFRASTRUCTURE

Terraform lets operators easily use the same configurations in multiple places to reduce mistakes and save time.

ENVIRONMENT PARITY

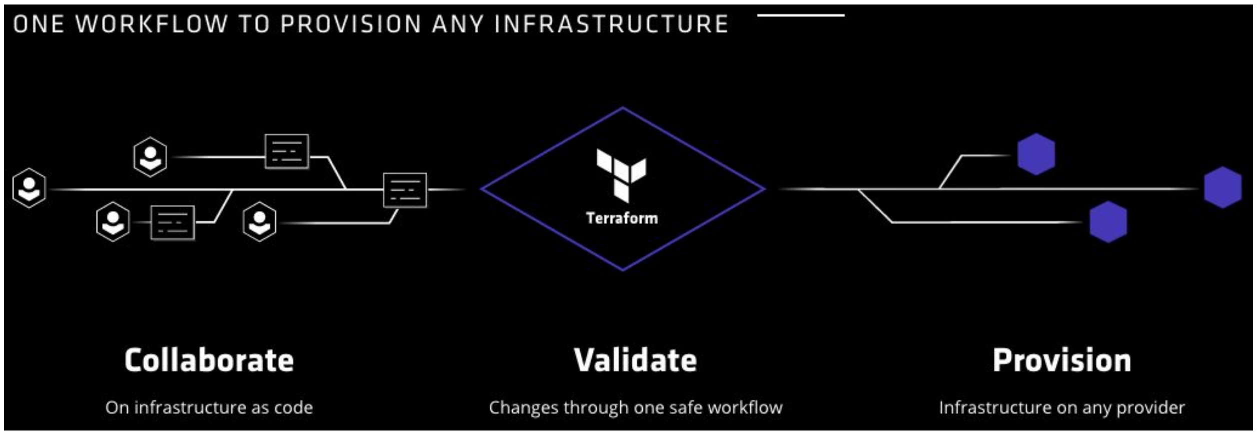
Use the same Terraform configuration to provision identical staging, QA, and production environments.

SHAREABLE MODULES

Common Terraform configurations can be packaged as modules and used across teams and organizations.

COMBINE MULTIPLE PROVIDERS CONSISTENTLY

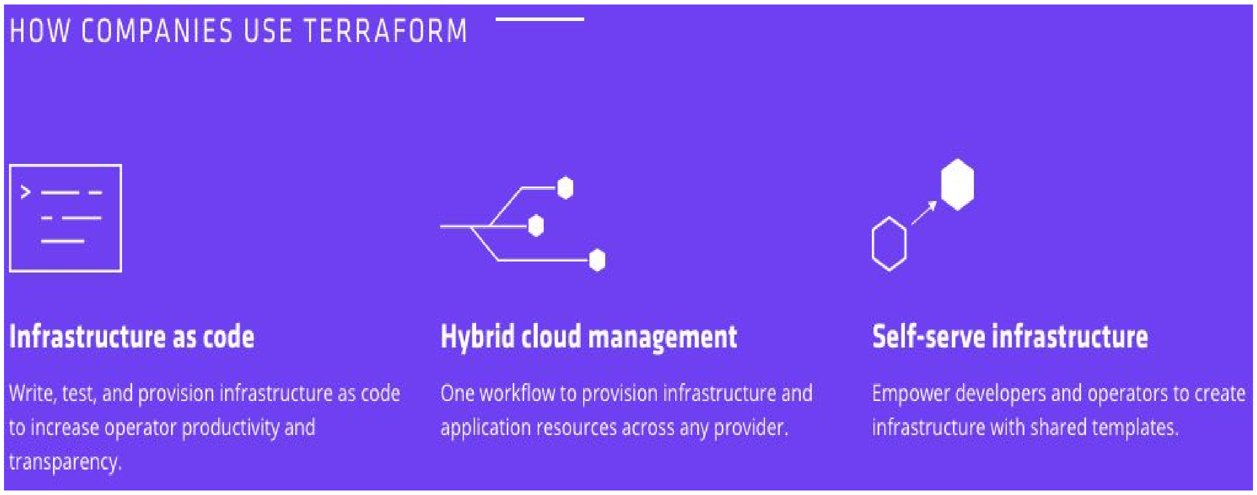
Terraform allows you to effortlessly combine high-level system providers. Launch a server from one cloud provider, add a DNS entry with its IP with a different provider. Built-in dependency resolution means things happen in the right order.


ONE WORKFLOW TO PROVISION ANY INFRASTRUCTURE

Collaborate
On infrastructure as code

Validate
Changes through one safe workflow

Provision
Infrastructure on any provider


HOW COMPANIES USE TERRAFORM

Infrastructure as code
Write, test, and provision infrastructure as code to increase operator productivity and transparency.

Hybrid cloud management
One workflow to provision infrastructure and application resources across any provider.

Self-serve infrastructure
Empower developers and operators to create infrastructure with shared templates.

nuaware

TERRAFORM FEATURES

Provision any infrastructure

Make safe, predictable, and transparent changes to any infrastructure

	Open Source	Pro	Enterprise
Infrastructure as code configuration	✓	✓	✓
Separation of infrastructure plans and applies	✓	✓	✓
Multi-provider support	✓	✓	✓

Usability and Collaboration

Safely make infrastructure changes across a team of users

	Open Source	Pro	Enterprise
Version control integration (GitHub, GitLab, Atlassian)		✓	✓
Remote plans, applies, and state storage		✓	✓
Environment locking		✓	✓
Rollback workflow		✓	✓
Audit log of all changes		✓	✓
Artifact build, storage, and versioning pipeline		✓	✓
Multi-environment workflow		✓	✓

Security and governance

Apply policy to all Terraform actions to meet compliance requirements

	Open Source	Pro	Enterprise
MFA workflow on AWS			✓
Private install options			✓

Support

	Open Source	Pro	Enterprise
Email Support		✓	✓
24x7 support with SLA			✓

nuaware

Vault

A Tool for Managing Secrets

Vault provides a centralized approach to secrets-management across every element of the application delivery lifecycle. Vault provides a highly available and secure way of storing and exposing secrets to applications and end users; for example, encryption keys, API tokens, and database credentials. Vault allows teams to consume the data they need without coordinating with security teams. Security teams can change passwords, rotate credentials, and update policies without coordinating across the organization.

Vault secures, stores, and tightly controls access to tokens, passwords, certificates, API keys, and other secrets in modern computing. Vault handles leasing, key revocation, key rolling, and auditing. Through a unified API, users can access an encrypted Key/Value store and network encryption-as-a-service, or generate AWS IAM/STS credentials, SQL/NoSQL databases, X.509 certificates, SSH credentials, and more.

FEATURES

Secret Storage

Vault can store your existing secrets, or it can dynamically generate new secrets to control access to third-party resources or provide time-limited credentials for your infrastructure. All data that Vault stores is encrypted. Any dynamically-generated secrets are associated with leases, and Vault will automatically revoke these secrets after the lease period ends. Access control policies provide strict control over who can access what secrets.

- General Secret Storage
- Employee Credential Storage
- API Key Generation for Scripts
- Data Encryption

Key Rolling

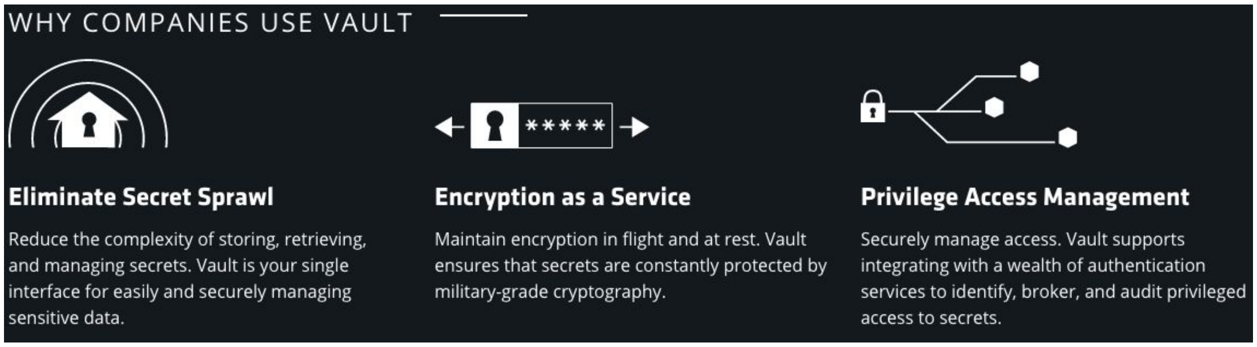
Secrets you store within Vault can be updated at any time. If using Vault's encryption-as-a-service functionality, the keys used can be rolled to a new key version at any time, while retaining the ability to decrypt values encrypted with past key versions. For dynamically-generated secrets, configurable maximum lease lifetimes ensure that key rolling is easy to enforce.

- Lease, Renew, and Revoke
- Lease IDs
- Lease Durations and Renewal
- Prefix-based Revocation

Audit Logs

Vault stores a detailed audit log of all authenticated client interaction: authentication, token creation, secret access, secret revocation, and more. Audit logs can be sent to multiple backends to ensure redundant copies. Paired with Vault's strict leasing policies, operators can easily trace the lifetime and origin of any secret.

- Audit Backends
- Sensitive Information
- Enabling/Disabling Audit Backends
- Blocked Audit Backends
- API



WHY COMPANIES USE VAULT

Eliminate Secret Sprawl

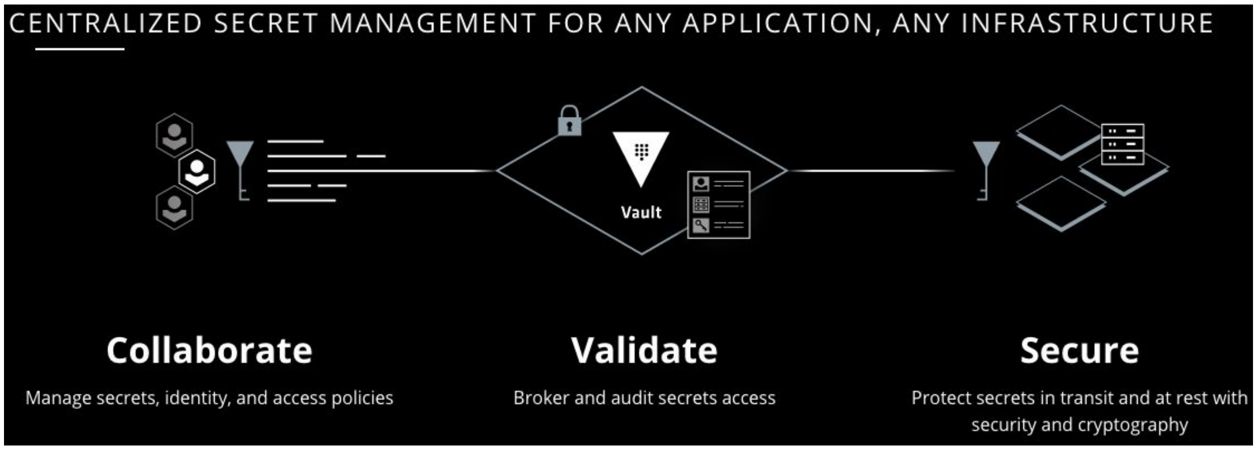
Reduce the complexity of storing, retrieving, and managing secrets. Vault is your single interface for easily and securely managing sensitive data.

Encryption as a Service

Maintain encryption in flight and at rest. Vault ensures that secrets are constantly protected by military-grade cryptography.

Privilege Access Management

Securely manage access. Vault supports integrating with a wealth of authentication services to identify, broker, and audit privileged access to secrets.


CENTRALIZED SECRET MANAGEMENT FOR ANY APPLICATION, ANY INFRASTRUCTURE
Vault
Collaborate
Manage secrets, identity, and access policies
Validate
Broker and audit secrets access
Secure
Protect secrets in transit and at rest with security and cryptography

VAULT FEATURES

Secrets, identity, and policy management

	Open Source	Pro	Enterprise
Secure secret storage, leasing and revocation	✓	✓	✓
Identity provider integrations	✓	✓	✓
Access control policies	✓	✓	✓
Detailed audit logs	✓	✓	✓
Key rolling	✓	✓	✓
Encryption as a service	✓	✓	✓

Usability and collaboration

	Open Source	Pro	Enterprise
UI for secret and policy editing		✓	✓
Health dashboard		✓	✓
Init and unseal workflow		✓	✓
Backup/restore		✓	✓

Operations & Support

	Open Source	Pro	Enterprise
Email support		✓	✓
24x7 phone support			✓
HSM Integration			✓
Multiple data center replication			✓

Nomad

Easily deploy applications at any scale

A Distributed, Highly Available, Datacenter-Aware Scheduler

Nomad is a cluster manager and scheduler. Schedulers allow an organization to decouple concerns even further, and abstract machines away from developers entirely. Instead, developers focus on the applications they want to run, and allow a scheduler to place the application and manage capacity of the machines. Nomad allows operators to provision a fleet of machines, decoupled from developers who are submitting jobs to Nomad. Nomad places the applications on available machines, allowing operators and developers to avoid manual coordination.

Nomad is a tool for managing a cluster of machines and running applications on them. Nomad abstracts away machines and the location of applications, and instead enables users to declare what they want to run and Nomad handles where they should run and how to run them. Key Features of Nomad include;

- Docker Support
- Operationally Simple
- Multi-Datacenter and Multi-Region Aware
- Flexible Workloads
- Built for Scale

Easily Deploy

Users submit simple high-level jobs and Nomad handles scheduling, deploying and upgrading applications.

Nomad makes it easy to deploy one container or thousands.

Any Cloud or Every Cloud

Multi-Datacenter and Multi-Region support enables running applications on any cloud, public or private.

Flexible Workloads

Nomad has extensible support for task drivers, allowing it to run containerized, virtualized, and standalone applications. Users can easily start Docker containers, VMs, or application runtimes like Java. Nomad supports Linux, Windows, BSD and OSX, providing the flexibility to run any workload.

Simplify Operations

Nomad simplifies operations by supporting blue/green deployments, automatically handling machine failures, and providing a single workflow to deploy applications

Increase Density, Reduce Cost

Nomad packs applications onto servers to maximize resource utilization, increase density, and reduce costs.

Consul

Service Discovery and Configuration Made Easy

Consul is HashiCorp's service discovery and monitoring tool. Consul allows applications that are running to broadcast their availability, and makes them easily reached by other applications. For example, web servers can use Consul to find their upstream databases or API services. Consul also monitors the health of applications to ensure only healthy instances receive traffic, and it notifies developers or operators of any issues. This allows development teams to avoid coordinating on IP addresses, and pushes discovery into the runtime of the application, so services can be updated independently. This ties into the rise of microservices and service-oriented architectures. Productivity is improved by having independent services updated by separate teams in parallel, without the kind of coordination required by a monolithic code base.

What is Consul?

Consul has multiple components, but as a whole, it is a tool for discovering and configuring services in your infrastructure. It provides several key features:

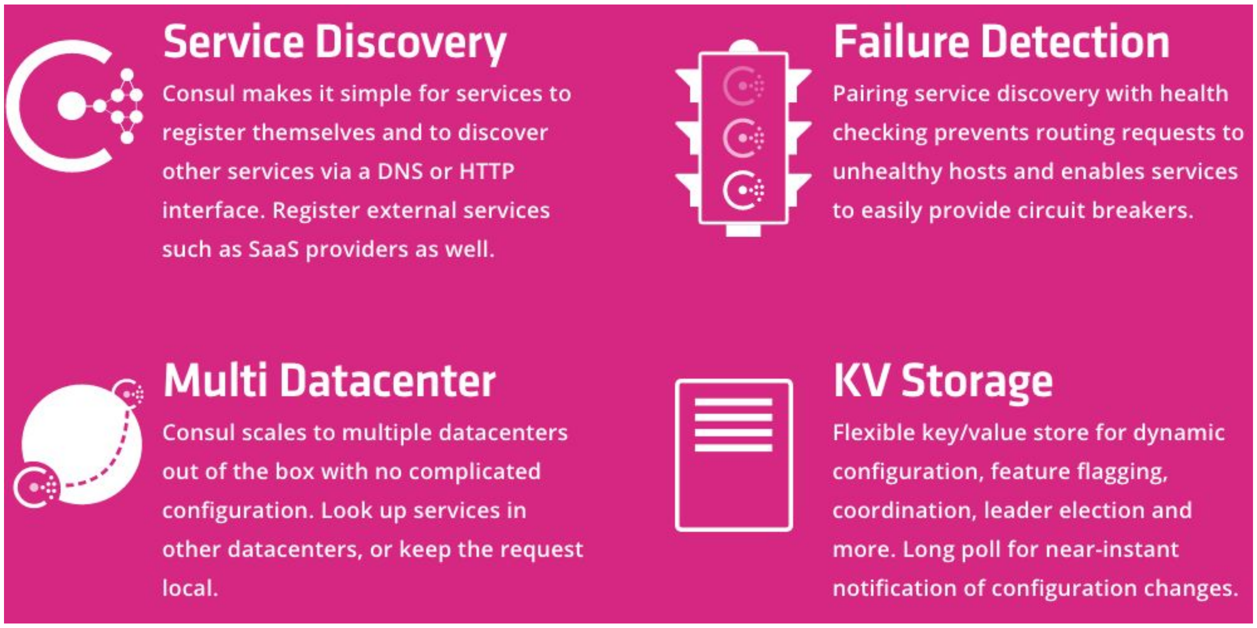
Service Discovery: Clients of Consul can provide a service, such as api or mysql, and other clients can use Consul to discover providers of a given service. Using either DNS or HTTP, applications can easily find the services they depend upon.

Health Checking: Consul clients can provide any number of health checks, either associated with a given service ("is the webserver returning 200 OK"), or with the local node ("is memory utilization below 90%"). This information can be used by an operator to monitor cluster health, and it is used by the service discovery components to route traffic away from unhealthy hosts.

KV Store: Applications can make use of Consul's hierarchical key/value store for any number of purposes, including dynamic configuration, feature flagging, coordination, leader election, and more. The simple HTTP API makes it easy to use.

Multi Datacenter: Consul supports multiple datacenters out of the box. This means users of Consul do not have to worry about building additional layers of abstraction to grow to multiple regions.

Consul is designed to be friendly to both the DevOps community and application developers, making it perfect for modern, elastic infrastructures.



Service Discovery

Consul makes it simple for services to register themselves and to discover other services via a DNS or HTTP interface. Register external services such as SaaS providers as well.

Failure Detection

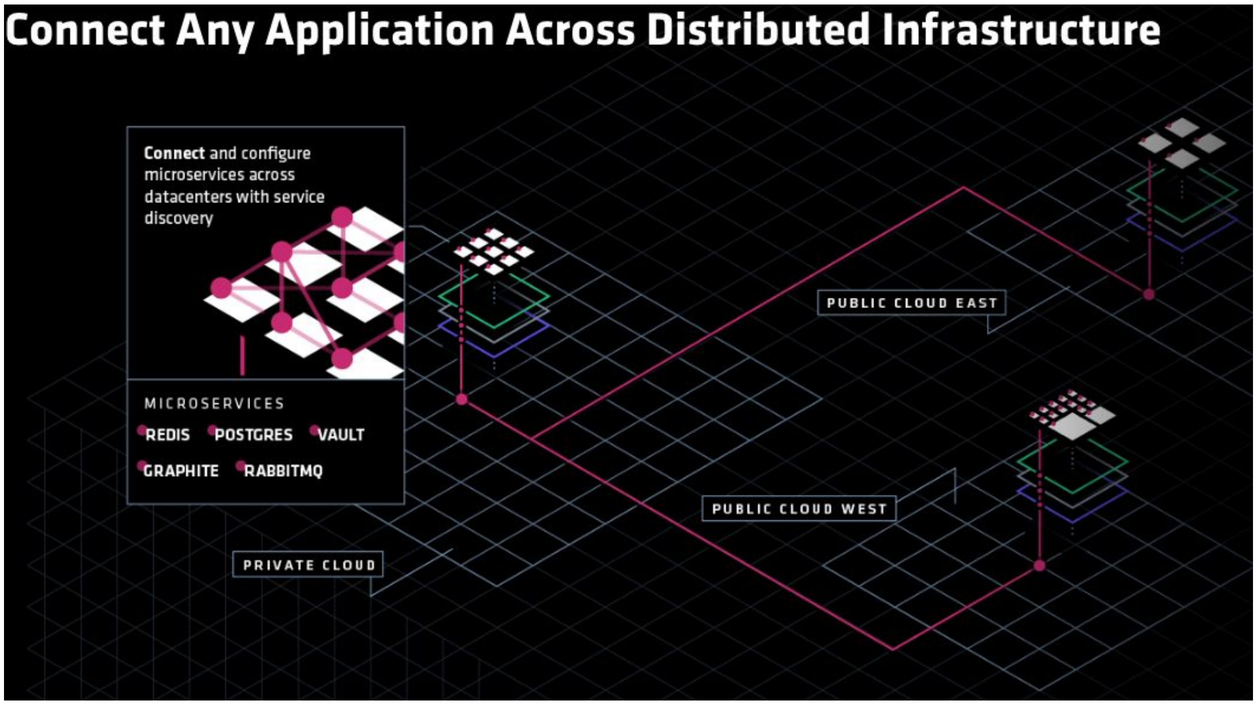
Pairing service discovery with health checking prevents routing requests to unhealthy hosts and enables services to easily provide circuit breakers.

Multi Datacenter

Consul scales to multiple datacenters out of the box with no complicated configuration. Look up services in other datacenters, or keep the request local.

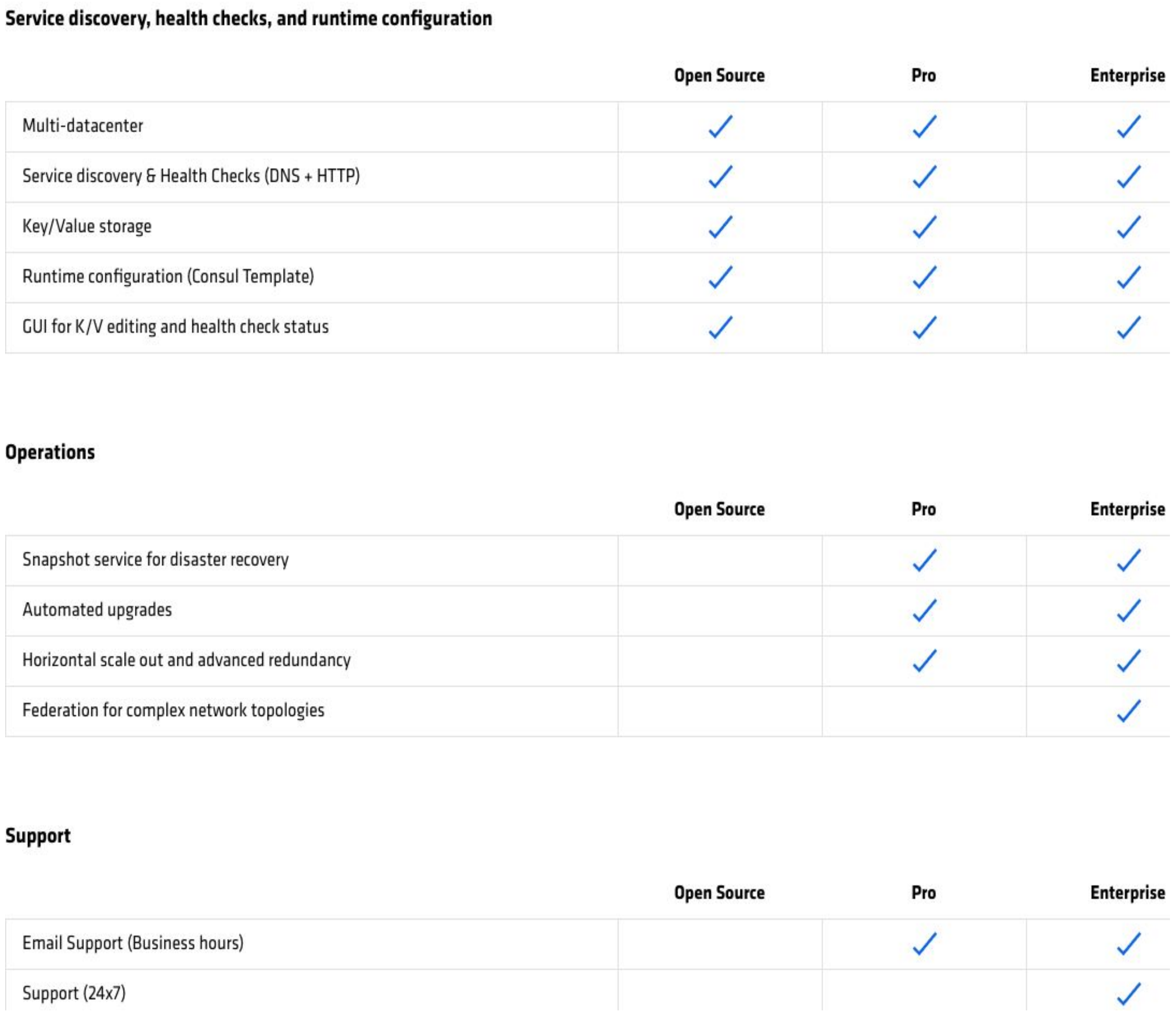
KV Storage

Flexible key/value store for dynamic configuration, feature flagging, coordination, leader election and more. Long poll for near-instant notification of configuration changes.


Connect Any Application Across Distributed Infrastructure
Connect and configure microservices across datacenters with service discovery
MICROSERVICES
REDIS
POSTGRES
VAULT
GRAPHITE
RABBITMQ
PRIVATE CLOUD
PUBLIC CLOUD EAST
PUBLIC CLOUD WEST

www.nuaware.com 0203 488 0530 info@nuaware.com

CONSUL FEATURES



Service discovery, health checks, and runtime configuration

	Open Source	Pro	Enterprise
Multi-datacenter	✓	✓	✓
Service discovery & Health Checks (DNS + HTTP)	✓	✓	✓
Key/Value storage	✓	✓	✓
Runtime configuration (Consul Template)	✓	✓	✓
GUI for K/V editing and health check status	✓	✓	✓

Operations

	Open Source	Pro	Enterprise
Snapshot service for disaster recovery		✓	✓
Automated upgrades		✓	✓
Horizontal scale out and advanced redundancy		✓	✓
Federation for complex network topologies			✓

Support

	Open Source	Pro	Enterprise
Email Support (Business hours)		✓	✓
Support (24x7)			✓

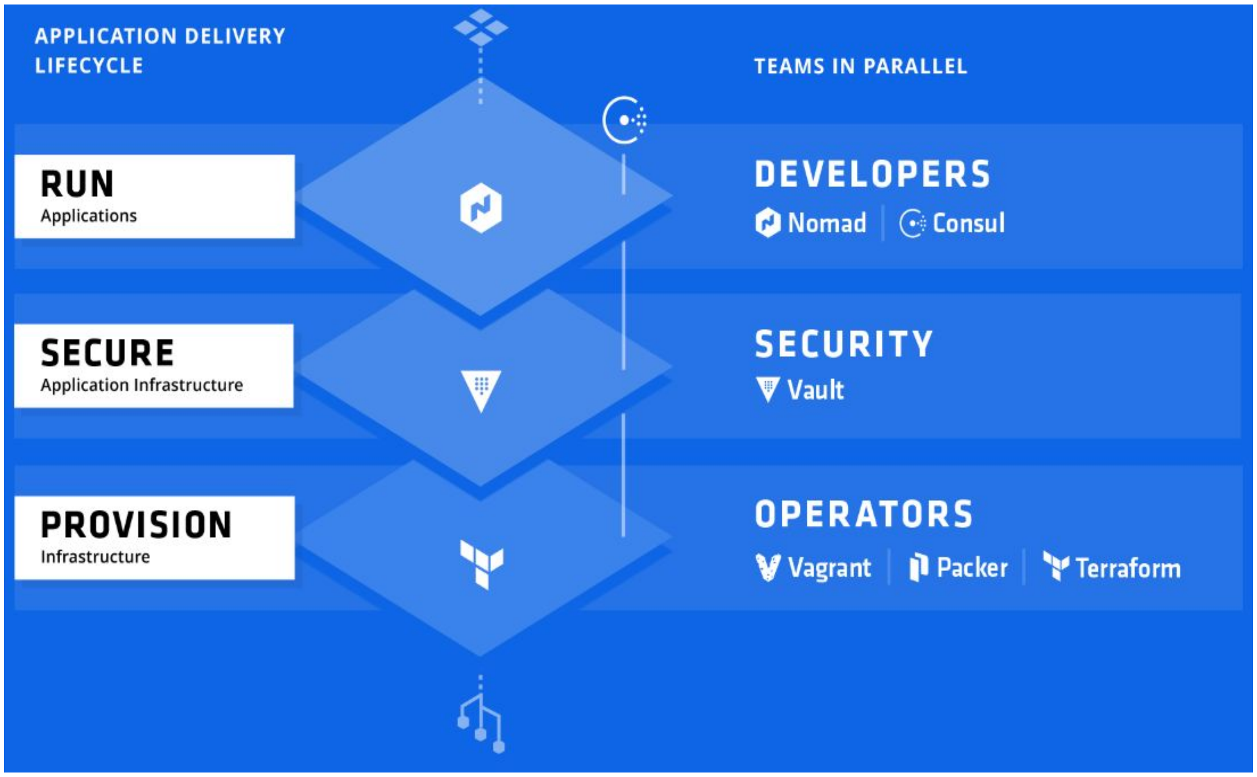
DevOps done right

Allowing Operations, Security and Development teams to work in parallel

As every company becomes a software company, the ability to execute a DevOps model allows them to deliver better applications, faster. And hundreds of thousands of software professionals globally are using the HashiCorp DevOps Suite to achieve this.

By providing a tool specifically designed for each of the elements of DevOps we allow the different participants in the software supply chain -- development, operations, and security -- to focus on their primary concern while unblocking their peers. This means turning what was a linear waterfall process into one where all three teams can run in parallel.

It is the parallelization of these workflows that is the essence of DevOps: the ability to Provision, Secure and Run any infrastructure for any application.+

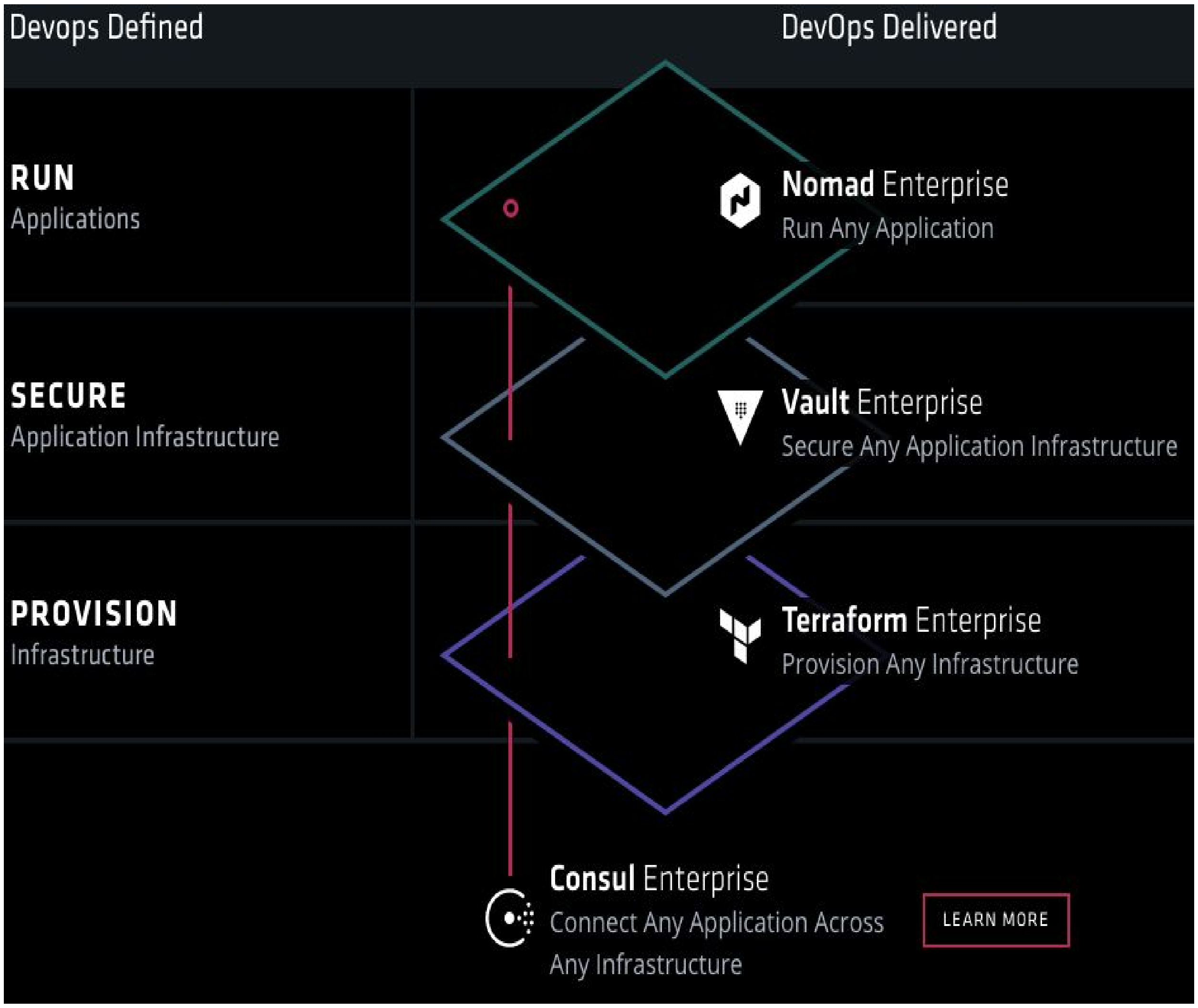


www.nuaware.com 0203 488 0530 info@nuaware.com

ENTERPRISE

HashiCorp Product Suite

The HashiCorp DevOps suite empowers organizations to provision hybrid infrastructure, secure secrets across distributed applications, and run dynamic resources for modern applications.



Devops Defined
DevOps Delivered
RUN
Applications
SECURE
Application Infrastructure
PROVISION
Infrastructure
Nomad Enterprise
Run Any Application
Vault Enterprise
Secure Any Application Infrastructure
Terraform Enterprise
Provision Any Infrastructure
Consul Enterprise
Connect Any Application Across Any Infrastructure
LEARN MORE