WHITE PAPER

RED HAT OPENSHIFT CONTAINER PLATFORM PRODUCT APPLICABILITY GUIDE FOR FISMA MODERATE

APPLICABILITY TO ASSIST CUSTOMERS IN FISMA MODERATE DEPLOYMENTS

JASON MACALLISTER
MITCH ROSS | CISSP
BRIAN JUSTICE | CISSP

VERSION 1.0

redhat.

North America | Europe
877.224.8077 | info@coalfire.com | Coalfire.com

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Red Hat, Inc. (Red Hat) delivers a comprehensive portfolio of products and services built from open source software components using an affordable, predictable subscription and support model. Red Hat engaged Coalfire, a respected cybersecurity engineering, advisory, and assessment company, to conduct an independent technical assessment of Red Hat OpenShift Container Platform (OpenShift) on Red Hat Enterprise Linux. The purpose of this product applicability guide is to identify the alignment of OpenShift on Red Hat Enterprise Linux to the Federal Information Security Management Act (FISMA) of 2014 security controls based on a Moderate impact-level categorization. Consideration for alignment is based on analysis of the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53 revision 4 Moderate impact security baseline and application container security guidance provided by NIST SP 800-190.

OpenShift is a container platform that natively integrates open source Linux container technologies and Kubernetes, combining them in an enterprise solution running on Red Hat Enterprise Linux. OpenShift provides an API, Web Console, and CLI to manage the underlying container technologies and Kubernetes to allow users to orchestrate the creation and management of containers. OpenShift provides self-service build and deployment automation for containers in addition to operational container features including scaling, monitoring, and management capabilities.

This product applicability guide may be useful for government agencies or other entities desiring to utilize container technologies within the framework of a FISMA Moderate impact security program of compliance. The guide discusses the relevant FISMA Moderate requirements that are applicable to OpenShift on Red Hat Enterprise Linux. The focus of this paper is on technical controls that are pertinent to and in alignment with OpenShift capabilities.

COALFIRE OPINION

Security controls, features, and functionality that are built into OpenShift on Red Hat Enterprise Linux can support and/or address relevant technical FISMA Moderate requirements as well as address relevant sections of NIST SP 800-190 as outlined in the compliance applicability detail section of this paper as it pertains to orchestration, management, monitoring, and scaling of containerized workloads.

INTRODUCING FISMA

FISMA is a United States federal law enacted in 2002 that mandates a process to strengthen the security posture of the federal government's information systems, bureaus, departments, and their supporting entities, such as vendors and subcontractors. When most agencies (and their vendors) discuss being "FISMA compliant," they are usually referring to meeting the controls identified in NIST SP 800-53 Rev 4. Federal government agencies, and their vendors and subcontractors that provide information systems to agencies, must prove, through annual assessment, that they meet FISMA requirements This is because the law is enforced through various processes (as described by the Office of Management Budget Circular [OMB] A-130), which establish definitions, processes, and requirements for federal agencies to follow. FISMA recommends guidance issued by NIST, such as FIPS 199, FIPS 200, NIST SP 800-53A, NIST SP 800-53 Rev 4, and so forth. The control selection, implementation, and testing are where IT professionals responsible for FISMA compliance perform most of the work. Meeting the compliance requirements is essential to receiving an authority to operate by government agencies.

UNDERSTANDING FISMA SCOPE

Beyond a selection of security controls, FISMA requires each agency to develop, document, and implement an agency-wide program to provide information security for the information and information systems that support the operations and assets of the agency, including those provided or managed by another agency, contractor, or other sources. The selection of security controls is part of a more comprehensive risk management framework, based on NIST SP 800-37. Figure 1 illustrates the major components of the risk management framework and the documents that guide each component.

Figure 1 - NIST Risk Management Framework

The selection of controls is based on categorization and assignment of impact relative to the category or categories of data being protected.

The goal of an agency's security program is to conduct day-to-day operations of the agency and to accomplish the agency's stated missions with adequate security, or security commensurate with risk, including the magnitude of harm resulting from the unauthorized access, use, disclosure, disruption, modification, or destruction of information.

Within NIST SP 800-53 Rev 4, baselines (low, moderate, high) have been established that determine the minimum set of requirements necessary to be authorized to manage data at these relative impact levels.

ADDITIONAL RELEVANT PUBLICATIONS

NIST has also published Special Publication 800-190 Application Container Security Guide. The purpose of this document is to explain the security concerns associated with container technologies and make practical recommendations for addressing those concerns when planning for, implementing, and maintaining containers. This publication is a guide providing recommendations to help ensure the security of container technology implementations and usage.

INTRODUCING OPENSHIFT CONTAINER PLATFORM

OpenShift is a comprehensive enterprise-grade application platform built for containers with Kubernetes. It is an integrated platform to run, orchestrate, monitor, and scale containers. OpenShift allows organizations to control, defend, and extend the application platform throughout an application's lifecycle. It enables a secure software supply chain to make applications more secure without reducing developer productivity and provides a consistent operations and management experience across any infrastructure in support of many teams.

OPENSHIFT CONTAINER PLATFORM ARCHITECTURE

The following is a list of components and roles that support OpenShift.

Operating System (OS) - OpenShift can be deployed on either Red Hat Enterprise Linux or, soon, Red Hat CoreOS.

Red Hat CoreOS is a container- and Kubernetes-optimized, minimal footprint OS powered by much of the same source as Red Hat Enterprise Linux. This pre-hardened OS will assist organizations with meeting requirements for least functionality due to its lightweight, purpose-built nature, as it only includes necessary features, functions, and services to host containers in an OpenShift environment.

Red Hat Enterprise Linux has built in security features and functionality that, as configured in an OpenShift installation, provide a secure platform for supporting the OpenShift components and the workloads in containers that OpenShift orchestrates.

Operating Environment - OpenShift can be deployed on bare-metal physical hardware, on virtual infrastructure, or in the cloud. It can be deployed on private or certified public cloud environments, depending on the organization's specific use cases.

OCI Runtime - OpenShift uses an Open Container Initiative (OCI)-compatible runtime for the execution of Linux containers. OCI is an open governance structure for the express purpose of creating open industry standards around container formats and runtime.

Kubernetes – Kubernetes provides orchestration for complex multi-container services. Kubernetes also provides scheduling for services across a container host cluster. To Kubernetes, OpenShift adds developer- and operations-centric tools that enable rapid application development, easy deployment and scaling, and long-term life-cycle maintenance for applications. OpenShift also leverages integrated components from Kubernetes to automate application builds, deployments, scaling, health management, and more. Included in the automation capabilities of OpenShift is the ability to configure and deploy Kubernetes container host clusters.

OpenShift Container Platform Components

The following components are specific to OpenShift itself.

OpenShift Nodes – Nodes are instances of Red Hat Enterprise Linux with the OpenShift software installed. Nodes are where end-user applications are ultimately run in containers. Nodes will contain the necessary OpenShift node daemon, the container runtime, and other necessary services to support the hosting of containers. Most of the software components that run above the OS (e.g., the software-defined network daemon) all run in containers themselves on the Node.

Containers – End-user application instances, application components, or other services are run in Linux containers. This OCI-compatible container technology provides an open source software development and

delivery platform that allows applications to be packaged for portability. The container only includes the necessary libraries, functions, elements, and code required to run the application.

Pod –While application components run in containers, OpenShift orchestrates and manages pods. A pod is an orchestrated unit in OpenShift made up of one or more containers. OpenShift will schedule and run all containers in a pod together on the same host. Generally, a pod should only contain a single function such as app server or web server and should not include multiple functions such as database and app server.

OpenShift Master – The Master is the control plane for OpenShift. The Master maintains and understands the state of the environment and orchestrates all activity that occurs on the Nodes. Just like the Nodes, the OpenShift Master is run on Red Hat Enterprise Linux. While the Master is technically also a Node and can participate in the software-defined network, for separation of function, the OpenShift Master should NOT be scheduled to run application instances (pods). The following are the four functions of the OpenShift Master:

> **API and Authentication** – The Master provides the single API that all tooling and systems interact with. Everything that interacts with OpenShift must go through this API. All API requests are SSL-encrypted and must be authenticated. Authorizations are handled by fine-grained role-based access control (RBAC). It is recommended to tie the Master to an external identity and access management system using LDAP, OAuth, or other providers. The Master evaluates requests for both authentication (AuthN) and authorization (AuthZ). Users of OpenShift who have been granted access can be authorized to work with specific projects.
>
> **Desired and Current State** – The state of OpenShift is held in the OpenShift data store. The data store uses etcd, a distributed key-value store. The data store houses information about the OpenShift environment and pertaining to the OpenShift Master, including user account information and the RBAC rules; the OpenShift environment state, including application environment information and non-application user data; and important environment variables, secrets data, and other information.
>
> **Scheduler** – The scheduler determines pod placement within OpenShift. It uses a combination of configuration and environment state (CPU, memory, and other environmental factors) to determine the best fit for running pods across the Nodes in the environment. The scheduler is configured with a simple JSON file in combination with Node labels to carve up OpenShift. This allows placement of pods within OpenShift to be based on the real-world topology, making use of concepts such as regions, zones, or other constructs relevant to the enterprise. These factors can contribute to the scheduled placement of pods in the environment and can ensure that pods run on appropriate Nodes associated with their function.
>
> **Health and Scaling** – The OpenShift Master is also responsible for monitoring the health of pods and scaling the pods as desired to handle additional load. The OpenShift Master executes liveness and readiness tests using probes that are defined by users. The OpenShift Master can detect failed pods and remediate failures as they occur.

Service Layer – The OpenShift Service Layer allows for application components to easily communicate with one another. For instance, a front-end web service containing multiple web servers would connect to database instances by communication via the database service. OpenShift automatically and transparently handles load balancing across the services' instances. In conjunction with probes, the OpenShift Service Layer ensures that traffic is only directed toward healthy pods, which helps to maintain component availability.

Persistent Storage – Linux containers are natively ephemeral and only maintain data for as long as they are running. Applications and/or application components may require access to a long-term persistent storage repository, such as may be required for a database engine. OpenShift provides the means to connect pods to external real-world storage, which allows for stateful applications to be used on the platform. Persistent storage types that are usable include iSCSI, Fiber Channel, and NFS, as well as cloud-type storage and software-defined storage options such as Red Hat OpenShift Container Storage. Persistent storage can be dynamically provisioned upon the user's request, provided the storage solution has an integration with OpenShift.

OpenShift Router – The routing layer provides external access to applications hosted in OpenShift. The routing layer operates in partnership with the Service Layer and provides automated load balancing to pods for external clients. The OpenShift Router runs in pods on the platform but receives traffic from the outside world and proxies the traffic to the appropriate pods. The OpenShift Router uses the service endpoint information to determine where to route and load balance traffic; however, it does not route traffic through the Service Layer.

OpenShift SDN – The OpenShift software-defined network (SDN) is a unified cluster network that enables communication between pods across the OpenShift cluster. The OpenShift SDN configures an overlay network that uses Open vSwitch (OVS). Red Hat currently provides three SDN plug-ins for use with OpenShift. The ovs-subnet plug-in provides a "flat" pod network where every pod can communicate with every other pod and service cluster-wide. The ovs-multitenant plug-in provides project-level isolation for pods and services. Each project receives a unique Virtual Network ID (VNID) that identifies traffic from pods assigned to the project. Pods from different projects cannot send packets to or receive packets from pods and services of a different project by default. Administrators of OpenShift can join or isolate projects as required. Lastly, the ovs-networkpolicy plug-in provides extremely fine-grained access control via user-defined rules. Network policy rules can be built in a "mandatory access control" style, where all traffic is denied by default unless a rule explicitly exists, even for pods/containers on the same host.

OpenShift Registry – The OpenShift Registry provides integrated storage and management for sharing container images, but OpenShift can utilize existing OCI-compliant container registries that are accessible to the Nodes and the OpenShift Master via the network.

Figure 2 is a high-level illustration of the OpenShift components.

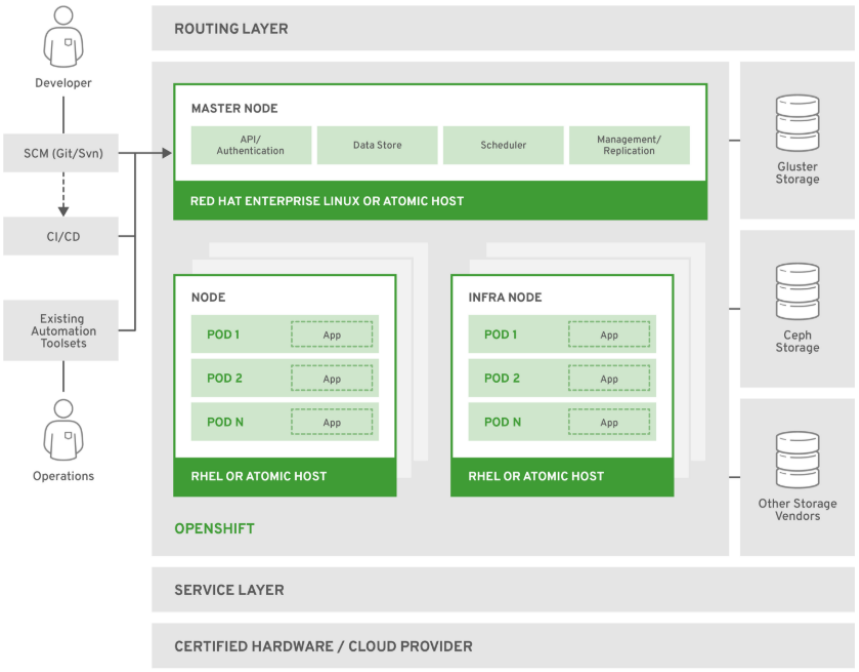

Figure 2: High-Level OpenShift Architecture

Users – User (operators, developers, application administrators) access to OpenShift is provided through standard interfaces including the Web UI, CLI, and IDEs. These interfaces go through the authenticated and RBAC-controlled API. Users do not require system-level access to any of the OpenShift hosts, even for complicated application debugging and troubleshooting tasks.

There are three types of users that can exist in an OpenShift environment: regular users, system users, and service accounts.

> **Regular users** are created automatically in the system upon first logon or via the API. Most interactive OpenShift users, including operators, developers, and application administrators, will be represented by this type of user account.
>
> **System users** This is the system:admin account which is a regular user type of account with elevated privileges. The system:admin account is the cluster administrator account that gets created when the system is setup for the first time. This account has special privileges and can only be logged on via a certificate from the console of the OpenShift Master.
>
> **Service accounts** are non-human system users, often associated with projects, used for API access in automation situations. Some default service accounts are created and associated when a project is first created. Project and cluster administrators can create additional service accounts for defining access to the contents of each project.

Projects – A project is a Kubernetes namespace with additional OpenShift annotations and metadata. It is the central vehicle by which access to resources for regular users is managed and is essentially the tenancy model of OpenShift and Kubernetes. A project allows a community of users to organize and manage their content in isolation from other communities.

For more information on OpenShift concepts, features, and functions, please refer to Red Hat's product documentation.

OPENSHIFT CONTAINER PLATFORM SECURITY

OpenShift enables continuous security with a defense-in-depth and secure software supply chain to the application platform. Security controls can be applied dynamically to the platform and the applications the platform supports. This allows security controls to keep up with the scale and agility of applications deployed in the platform. OpenShift runs on Red Hat Enterprise Linux and makes heavy use of the existing security features built into the OS. Red Hat manages the OS packages and provides trusted distribution of content. Red Hat is committed to responsive action to security vulnerabilities. The security of OpenShift includes and utilizes hardened technologies such as SELinux; process, network, and storage separation; proactive monitoring of capacity limits (CPU, disk, memory, etc.); and encrypted communications for infrastructure support including SSH, SSL, etc. Additionally, OpenShift provides integration with third-party identity management solutions to support secure authentication and authorization options in alignment with organization compliance requirements.

The following is a high-level list of OpenShift security features and capabilities.

Container Host and Platform Multitenancy – Red Hat Enterprise Linux can manage multitenancy for the container runtime by using Linux namespaces, SELinux, CGroups, and Secure Computing Mode (seccomp) to isolate and protect containers, which can be useful for maintaining separation for workloads of differing classifications.

Container Content – The Red Hat Container Catalog delivers validated application content from Red Hat and certified ISV partners.

Container Registries – OpenShift includes an integrated container registry that provides basic functionality supporting build and deployment automation within the cluster, tied into the OpenShift RBAC. Within the context of an organization needing to adhere to FISMA Moderate requirements, Red Hat Quay is an additional product that provides a registry with capabilities for both RBAC and vulnerability scanning of applications and software in images and more.

Building Containers – OpenShift integrates tightly with Jenkins and can be easily integrated with other Continuous Integration/Continuous Delivery (CI/CD) tools to manage builds, code inspection, code scanning, and validation.

Deploying Containers – By default, OpenShift prevents containers from running as root or other specifically-named users. In addition, OpenShift enables granular deployment policies that allow operations, security, and compliance teams to enforce quotas, isolation, and access protections.

Container Orchestration – OpenShift integrates secure operational capabilities to support trust between users, applications, and security policies.

Network Isolation – OpenShift uses a SDN approach to provide a unified cluster network that enables communication between pods across the OpenShift cluster. The pod network is established and maintained by the OpenShift SDN plug-ins, which create an overlay network using OVS. There are three SDN plug-ins available from Red Hat as options for the customer to deploy: the ovs-subnet, ovs-multi-tenant, and ovs-networkpolicy. Other third-party SDN solutions exist that are capable of being integrated into OpenShift.

Secure the data – OpenShift provides access to and integration with a broad range of storage platforms and protocols, allowing applications to securely store and encrypt application data.

API management – OpenShift can be integrated with the 3scale API Management platform to authenticate, secure, and rate-limit API access to applications and services.

SCOPE AND APPROACH FOR REVIEW

The understanding of OpenShift and Red Hat Enterprise Linux and their combined capabilities was gained through product specification, installation, configuration, administration, and integration documentation provided by Red Hat and generally made available from Red Hat's public-facing web site. Coalfire further conducted interviews and engaged in live product demonstrations with Red Hat personnel. For live product demonstration purposes, OpenShift was also implemented on Red Hat Enterprise Linux in a lab environment to provide hands-on testing and analysis of the system's capabilities to support compliance.

Coalfire's review of OpenShift on Red Hat Enterprise Linux began with a general alignment of the applicability of the technology against the high-level FISMA Moderate impact baseline control objectives. This was further narrowed down to specific requirements that were considered applicable to either OpenShift or the underlying OS. An analysis of capability for the reviewed technology to address the applicable requirements was then conducted. This analysis primarily focused on what an assessor might review when following the FISMA Moderate testing guide during an assessment of applicable requirements.

SCOPE OF TECHNOLOGY AND SECURITY STANDARD TO REVIEW

Coalfire was tasked by Red Hat to review OpenShift as deployed on Red Hat Enterprise Linux. The primary focus of the review included the components, features, and functionality of OpenShift along with the supporting underlying OS features and functionality when the components are deployed on Red Hat Enterprise Linux. Coalfire did not include in the assessment application pods or containers that an organization may deploy on OpenShift. Containers that were deployed in the lab environment were strictly used for the purposes of demonstrating the platform's orchestration, deployment, and management capabilities. Furthermore, Coalfire did not assess available image registries or repositories that may be used for acquiring applications, services, dependencies, or other elements to be hosted on or used within OpenShift.

For this review, Coalfire included requirements from NIST Special Publication 800-53 Revision 4 Security and Privacy Controls for Federal Information Systems and Organizations April 2013 publication available from https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf. For a broader understanding of the requirements and their applicability to technical solution implementation, Coalfire also reviewed supporting documentation provided by NIST including assessment guidance. Applied understanding of NIST 800-53 Rev. 4 requirements and guidance was supplemented by documentation and guidance on relevant subjects, many of which are referenced in the NIST SP 800-53 Rev. 4 requirements. As OpenShift is a container platform, Coalfire also applied guidance from NIST Special Publication 800-190 Application Container Security Guide September 2017 publication.

COALFIRE EVALUATION METHODOLOGY

Coalfire initially examined the FISMA Moderate impact baseline requirements and identified them as either procedural (organizational) or technical (implementation). Qualification of a requirement as procedural or technical was based on a review of the requirement narrative, testing procedures, and guidance.

"Non-technical" procedural requirements that include definition and documentation of policies, procedures, and standards were not considered directly applicable to the technical solution. Likewise, "non-technical" requirements including operational procedures that describe manual processes were not assessed against the technology's capability. Examples of this type of "non-technical" requirements included maintenance of

facility visitor logs, verification of an individual's identity prior to granting physical or logical access, performance of periodic physical asset inventories, or generation of network topology or flow diagrams.

Technical requirements were then assessed to determine applicability to the solution and/or solution components. Where achievement of the requirement objectives was more likely to be met using an external and non-adjacent mechanism, the requirement was determined to be "not applicable" to the assessed technology. Examples of requirements that Coalfire determined to be "not applicable" to OpenShift on Red Hat Enterprise Linux included the use of encryption key management, wireless networking, technical physical access controls, and antivirus solutions. That is not to say that these are not important factors to consider as it pertains to OpenShift, but rather that OpenShift does not natively provide these capabilities to the extent necessary to achieve compliance.

Where the requirement was qualified as applicable, Coalfire further assessed the capability of the solution to address the requirement. For applicable requirements, Coalfire designated a qualitative category of capability including whether the solution was determined to fully support the requirement, partially support the requirement, or unable to support the requirement. In cases where the requirement was determined to be applicable but unsupported, additional thought for the use of third-party solutions could be considered.

OPENSHIFT CONTAINER PLATFORM APPLICABILITY TO FISMA MODERATE

The table that follows details controls where OpenShift has applicability to address. This applicability applies either as a customer configurable item within OpenShift or as a native and default capability to address the control requirement.

Overall, there are general statements of applicability that can be made for OpenShift for each of the FISMA Moderate control families.

For the Access Control (AC) and Identity and Authentication (IA) families of controls, OpenShift will primarily be dependent on an external third-party identity and authentication provider. OpenShift provides several integration options for third-party identity and authentication services. Users and groups would be created in the external identity provider directory and assigned appropriate authenticators. The external provider would also be responsible for managing authenticators. Users would first be authenticated by the external provider prior to gaining access to the OpenShift API, web console, or CLI. The user is then tied to an identity within OpenShift for authorization. OpenShift provides an OAuth server to enable this integration. The customer should determine which option for identity and authentication best suits their use case with the understanding that some options may fit better within the context of a security program intended to address FISMA Moderate impact level controls baseline.

Users and groups established within the third-party identity and authentication service can be assigned RBACs within OpenShift. Administrators can use the CLI or web console to view RBAC resources and manage the roles and bindings. Roles can be used to grant various levels of access both cluster-wide and at the project-scope. Users and groups can be bound to multiple roles at the same time. The customer should carefully plan the implementation of OpenShift regarding assigning roles and responsibilities to individuals and groups. It will be important for the organization to ensure that there is not unbounded administrative access provided to the container platform. The understanding of application of access controls to the platform and to the projects will be important for ensuring that users or systems do not gain unauthorized access.

Awareness and Training (AT) requirements are also important to the security of OpenShift. The customer should make sure that all personnel responsible for managing and using OpenShift (administrators,

developers, operations personnel) be trained in both the proper design, implementation, and management of OpenShift and the operational nuances that are specific to the customer's security program. The awareness of training of personnel engaged with OpenShift should allow the community of users to be better equipped to identify potential issues with implementation and operation that could pose increased risk.

The customer will be responsible for identifying auditable events (AU) relevant to OpenShift as well as for the projects and containers that are orchestrated by the platform. While OpenShift has the capability to produce audit logs with records necessary to support FISMA Moderate requirements, the handling of audit logs and events will best be served by a third-party tool that collects or aggregates logs from multiple sources and correlates activities to identify security events from a wider perspective than just the container platform. The customer will also be responsible for coordinating the log generation capabilities of the workloads that are deployed on the platform.

The customer should also consider the impacts and relevance of OpenShift on their overall security program. Customers that are newly implementing OpenShift will want to make sure that the culture and methods of containerization are taken into consideration when evaluating their Security Assessment and Authorization (CA) policies and processes. Traditional approaches to security have been less and less relevant to information systems as they become more agile, portable, flexible, scalable, and dynamic. As layers of the information systems are becoming increasingly abstracted, it will be important for the organization to implement hardening tactics to reduce the surface areas of attack. Securing the containers themselves becomes increasingly challenging to traditional security methods, as containers and container environments do not often provide as concrete a security boundary as physical and virtual machines.

Proper Configuration Management (CM) techniques relative to the DevOps nature of containerization will also be important to the security of the environment. The organization will need to consider the software development lifecycle and security development lifecycle as it pertains to the use of OpenShift. Gateways should be clearly identified and managed whereby applications or application components are continually developed, tested, and deployed into production. Testing should be inclusive of both application function and application security to limit the possibility of vulnerable code, libraries, or other application components from being introduced to the environment. Once tested and approved, images from which containers are spawned should be digitally signed to prevent tampering. The organization should establish techniques to approve and establish baselines and standards by which containers are measured and implemented in the environment. For defining, documenting, selecting and implementing configuration management policies, procedures, standards, controls and management tools, the organization should consider the rapid and dynamic rate of change that can occur in container environments.

The portable nature of containers can be useful to organizations' Contingency Planning (CP) efforts. The organization should consider the components of the container platform that are necessary for recovery in case of failure and plan accordingly.

The customer should also consider the implications of risk associated with implementation and operationalization of OpenShift including the deployment of workloads or applications on the platform. A healthy understanding of risk will help the customer properly implement mitigating controls or safeguards to reduce the residual risk to an acceptable level. The customer should include the selection and implementation of controls for OpenShift and the applications it supports into their system security planning and establish action items and milestones for continuous improvement of security per the Risk Assessment (RA) family of requirements.

The customer should understand the commitment that Red Hat has to its customers regarding its development of enterprise class software. As part of the System and Services Acquisition (SA), the

customer is responsible for ensuring that their vendors are following best practices for the development of the systems that they provide to their customers. It will be important for the customer to address discovered vulnerabilities and apply security patches as Red Hat provides them in a timely manner.

Coalfire recommends that OpenShift be implemented within the defined external FISMA Moderate authorization boundary with edge protections (routers, firewalls, IDS/IPS, WAF, and so forth as necessary and/or required) between the Internet and out of scope systems and the in-scope systems. For east-west communications internal to the container environment, OpenShift provides built-in SDN capabilities to provide the management, control, and security of the data plane. This allows the organization to define internal boundaries between applications and/or application components. The customer should consider what is necessary to achieve the security posture that aligns with their program of compliance.

The customer should consider approaches to System and Information Integrity (SI) requirements that are relevant to the nature of container environments and their workloads. Traditional tool sets, e.g. antivirus protections, may not apply naturally to OpenShift or the supported workloads. The customer should choose controls, including tools that enable prevention, detection, and correction, that are designed for containerized workloads and their infrastructures. To help protect workloads, OpenShift can be configured to selectively orchestrate workloads such that workloads of differing sensitivity levels run on different Nodes. This reduces the possibility of a lower security workload compromising a higher security workload. Additionally, OpenShift has the capability of utilizing security features of Red Hat Enterprise Linux. For example, SELinux is enabled by default and containers are deployed with appropriate SELinux contexts to isolate the container's processes and prevent a rogue or exploited container from negatively impacting other containers, their persistent storage, or the container host.

OPENSHIFT CONTAINER PLATFORM FISMA MODERATE APPLICABILITY DETAIL

The following table provides detailed alignment of OpenShift's capabilities for alignment and to address FISMA Moderate control requirements. The requirements that are listed are the requirements that Coalfire determined to be applicable for OpenShift. Other FISMA requirements that are not listed in the following table are either not applicable to the technology or are intended to be addressed as a control response by the customer rather than the information system.

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
AC-3	ACCESS ENFORCEMENT	The information system enforces approved authorizations for logical access to information and system resources in accordance with applicable access control policies.	OpenShift uses granular RBAC for users and/or groups of users. RBAC objects determine whether a user can perform a given action on the system, based on the user's assigned role(s). This allows platform administrators to use cluster roles and bindings to control who has various access levels to OpenShift itself and all contained projects and resources and allows developers to use local roles and bindings to control who has access to their projects. Authorization is managed using rules, roles, and bindings. Rules are a set of permitted verbs (actions) on a set of objects, for example, whether something or someone can create pods. Roles are a collection of rules.

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
			Users and groups can be associated with roles or bound to multiple roles at the same time. Bindings are associations between users and/or groups with a role.
AC-4	INFORMATION FLOW ENFORCEMENT	The information system enforces approved authorizations for controlling the flow of information within the system and between interconnected systems based on [Assignment: organization-defined information flow control policies].	Coalfire recommends placing OpenShift on the inside of the external boundary of the overall information system with boundary protections (firewalls, routers, application gateways, web proxies, web gateways, IDS/IPS) provided externally to OpenShift. In this way, the information flow control to and from the Internet or uncontrolled network is provided by the organization's third-party solution (firewalls, IDS/IPS, routers, application gateways, web proxies, web gateways, and so forth). For communication between endpoints within OpenShift, container networking capabilities are provided to control the flow of information within the container environment and between pods. Kubernetes tells a Node to attach a pod to a network. The network plugin then allocates the pod an IP address from an internal network. This ensures that all containers within the pod behave as if they are on the same host and causes all containers within a pod to share a network space. Giving each pod its own IP address means that pods can be treated like physical hosts or virtual machines in terms of port allocation, networking, naming, service discovery, load balancing, application configuration, and migration. OpenShift uses a SDN approach to provide a unified cluster network that enables communication between pods across the OpenShift cluster. This pod network is established and maintained by one of the OpenShift SDN plug-ins, which configures an overlay network using OVS. Almost all packet delivery decisions are performed with OpenFlow rules in the OVS bridge br0, which provides flexible routing. Depending on the deployed SDN plug-in, network isolation can be enforceable and finely controlled. Pod networks can be isolated using network namespaces with OpenShift's multitenant-sdn plug-in. While each pod gets its own IP and

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
			port range to bind to, the multitenant-sdn plug-in ensures each pod project has its own virtual network within the SDN, thereby isolating pod networks from each other even on the same Node. The default behavior for the multitenant-sdn plug-in is that pods from different projects cannot send packets to or receive packets form pods and services of a different project. This can be used to isolate development, test, and production environments within a cluster.
AC-6 (10)	PROHIBIT NON-PRIVILEGED USERS FROM EXECUTING PRIVILEGED FUNCTIONS	The information system prevents non-privileged users from executing privileged functions to include disabling, circumventing, or altering implemented security safeguards/countermeasures.	OpenShift can prevent non-privileged users from executing privileged functions to include disabling, circumventing, or altering implemented security safeguards/countermeasures. Interaction with OpenShift is associated with a user. An OpenShift user object represents an actor that may be granted permissions in the system by adding roles to them or to their groups. The users that can exist include regular users and service accounts. OpenShift can be configured for RBAC. RBAC allows for granular determination of access for users and/or groups of users. RBAC objects determine whether a user can perform a given action system-wide. This allows platform administrators to use cluster roles and bindings to control who has various access levels to OpenShift itself and all contained projects. It allows developers to use local roles and bindings to control who has access to their projects. Authorization is managed using rules, roles, and bindings. Rules are a set of permitted verbs (actions) on a set of objects, for example, whether something or someone can create pods. Roles are a collection of rules. Users and groups can be associated with roles or bound to multiple roles at the same time. Bindings are associations between users and/or groups with a role.
AC-8	SYSTEM USE NOTIFICATION	The information system: a. Displays to users [Assignment: organization-defined system use notification message or banner] before granting access to the system that	Typically, users accessing the OpenShift Web Console, CLI, or API would be doing so from an organizationally controlled machine where system use notifications would be presented

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
		provides privacy and security notices consistent with applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance and states that: a.1. Users are accessing a U.S. Government information system; a.2. Information system usage may be monitored, recorded, and subject to audit; a.3. Unauthorized use of the information system is prohibited and subject to criminal and civil penalties; and a.4. Use of the information system indicates consent to monitoring and recording; b. Retains the notification message or banner on the screen until users acknowledge the usage conditions and take explicit actions to log on to or further access the information system; and c. For publicly accessible systems: c.1. Displays system use information [Assignment: organization-defined conditions], before granting further access; c.2. Displays references, if any, to monitoring, recording, or auditing that are consistent with privacy accommodations for such systems that generally prohibit those activities; and c.3. Includes a description of the authorized uses of the system.	to the user relative to access to the enterprise wide network. The organization is responsible for establishing the content of system use notifications and defining where the system use notifications would be displayed. The choice in where system use notifications is displayed should take into considerations the requirement to retain the notification banner on the screen until the user acknowledges the usage conditions and takes explicit action to logon. For a FISMA deployment, it is not recommended to make the OpenShift API, CLI, or Web Console directly publicly available. The OpenShift Web Console, though capable of being customized to provide the opportunity for the organization to display a system use notification, cannot enforce display prior to logon. The external authentication provider would more likely provide the system use notification with a flow control that requires acceptance prior to logon or would notify the user that logon indicates acknowledgement and acceptance of the system use notification.
AC-11	SESSION LOCK	The information system: a. Prevents further access to the system by initiating a session lock after [Assignment: organization-defined time period] of inactivity or upon receiving a request from a user; and b. Retains the session lock until the user reestablishes access using established identification and authentication procedures.	OpenShift provides the capability to enable a "soft" setting for inactivity/idle timeout. The soft setting is in masterConfig.oauthConfig.tokenConfig.accessTokenInactivityTimeoutSeconds. A hard setting option is available with the token expiration options in sessionConfig – sessionMaxAgeSeconds, which controls the maximum age of a session after which the session token expires and the user must login again.

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
AC-11 (1)	PATTERN-HIDING DISPLAYS	The information system conceals, via the session lock, information previously visible on the display with a publicly viewable image.	This is primarily applicable to the Web Console and CLI. After a specified period of idle time or when the user voluntarily disconnects, the session is terminated. The user would then be required to re-authenticate prior to further interaction with the interfaces.
AC-12	SESSION TERMINATION	The information system automatically terminates a user session after [Assignment: organization-defined conditions or trigger events requiring session disconnect].	The organization can define session idle timeouts as well as session timeouts whereupon re-authentication is required. The organization will be responsible for defining events or conditions requiring session termination. Through integration with the external authentication and authorization provider, the information system should be able to respond accordingly by terminating sessions for events such as users being removed from groups that previously authorized access and/or for user's accounts being disabled or deleted from the directory.
AU-3	CONTENT OF AUDIT RECORDS	The information system generates audit records containing information that establishes what type of event occurred, when the event occurred, where the event occurred, the source of the event, the outcome of the event, and the identity of any individuals or subjects associated with the event.	OpenShift is capable of supporting audit requirements by generating auditable events that contain information to establish what type of event occurred, when the event occurred, where the event occurred, the source of the event, the outcome of the event, and the identity of any individuals or subjects associated with the event.
AU-3 (1)	ADDITIONAL AUDIT INFORMATION	The information system generates audit records containing the following additional information: [Assignment: organization-defined additional, more detailed information].	OpenShift is not applicable to the selection of additional auditable events by the organization; however, there is an expectation that a request for additional auditable events would be able to be supported by OpenShift as requested.
AU-8	TIME STAMPS	The information system: a. Uses internal system clocks to generate time stamps for audit records; and b. Records time stamps for audit records that can be mapped to Coordinated Universal Time (UTC) or Greenwich Mean Time (GMT) and meets [Assignment: organization-defined granularity of time measurement].	OpenShift is dependent on time synchronization to support sensitive operations, such as log keeping and time stamps. OpenShift uses the time of the underlying host. OpenShift documentation provides guidance for enabling NTP synchronization for the underlying hosts in support of keeping consistent time among components of OpenShift. The host can be configured for coordinated time synchronization in support of time stamps using Coordinated Universal Time (UTC). OpenShift operations include etcd leader election and health checks for pods and some

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
			other issues and help prevent time skew problems.
AU-12	AUDIT GENERATION	The information system: a. Provides audit record generation capability for the auditable events defined in AU-2 a. at [Assignment: organization-defined information system components]; b. Allows [Assignment: organization-defined personnel or roles] to select which auditable events are to be audited by specific components of the information system; and c. Generates audit records for the events defined in AU-2 d. with the content defined in AU-3.	OpenShift can generate audit records and events pertaining to actions taken by users and/or system components against the API, CLI, or Web Console interface. OpenShift also generates log data and performance metrics relative to its normal operation. The organization is responsible for determining the capabilities of the information system for auditing events that are defined by the organization. Once those capabilities are understood, the organization is then responsible for ensuring that OpenShift is configured properly to generate the required and defined logs and events. It is recommended that selection of auditable events will be managed through an external syslog and/or SIEM solution. The organization will need to determine how the audit records are to be generated and how they meet the requirements defined in AU-2d and in AU-3.
CP-9	INFORMATION SYSTEM BACKUP	The organization: a. Conducts backups of user-level information contained in the information system [Assignment: organization-defined frequency consistent with recovery time and recovery point objectives]; b. Conducts backups of system-level information contained in the information system [Assignment: organization-defined frequency consistent with recovery time and recovery point objectives]; c. Conducts backups of information system documentation including security-related documentation [Assignment: organization-defined frequency consistent with recovery time and recovery point objectives]; and d. Protects the confidentiality, integrity, and availability of backup information at storage locations.	The organization can perform backups of OpenShift to save state to a separate storage. Red Hat provides documentation detailing methods for back up and restoration of OpenShift.

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
IA-6	AUTHENTICATOR FEEDBACK	The information system obscures feedback of authentication information during the authentication process to protect the information from possible exploitation/use by unauthorized individuals.	OpenShift natively obscures authentication information during the authentication process.
RA-5 (5)	PRIVILEGED ACCESS	The information system implements privileged access authorization to [Assignment: organization-identified information system components] for selected [Assignment: organization-defined vulnerability scanning activities].	Container scanning tools can leverage continuously updated vulnerability databases to ensure that the customer always has the latest information on known vulnerabilities for container content. OpenShift provides a pluggable API to support multiple scanners. Moreover, Red Hat Quay, an optional container registry sold by Red Hat, can be used to scan container images for known vulnerabilities. OpenShift enables the leveraging of such scanners with the CI/CD process. Static code analysis tools can be integrated to test for security flaws in source code and software composition analysis tools that identify open source libraries to provide metadata on those libraries including known vulnerabilities. OpenShift makes use of object annotations to extend functionality. External tools, such as vulnerability scanners, may annotate image objects with metadata to summarize results and control pod execution.
SC-4	INFORMATION IN SHARED RESOURCES	The information system prevents unauthorized and unintended information transfer via shared system resources.	OpenShift makes use of features of the underlying OS to isolate the workloads to their own processor, memory, and storage spaces. Unless the container is authorized to access or share a specific resource space, the container should not be able to access information unless authorized. In addition to the RBAC resources that control what a user can do, OpenShift provides security context constraints (SCCs) that control the actions that a pod can perform and what it can access. SCCs are objects that define a set of conditions that a pod must adhere to be accepted into the system. They allow an administrator to control: running of privileged containers, capabilities a container can request to be added, use of host directories as volumes, the SELinux context of the container, the user ID, the use of host namespaces and networking, allocating an FSGroup that owns the pod's volumes, configuring allowable supplemental groups, requiring the use of read only root file system,

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
			controlling the usage of volume types, and configuring allowable seccomp profiles. SCCs are useful for managing access to host storage and can be managed by administrators. Additionally, seccomp can be used to restrict the set of system calls that applications can make. Sysctl settings are exposed via Kubernetes, allowing users to modify certain kernel parameters at runtime for namespaces within a container. Only sysctls that are namespaced can be set independently on pods. If a sysctl is not namespaced, called node-level, it cannot be set within OpenShift. Moreover, only those sysctls considered safe are whitelisted by default; the customer can manually enable other unsafe sysctls on the Node to be available to the user.
SC-7	BOUNDARY PROTECTION	The information system: a. Monitors and controls communications at the external boundary of the system and at key internal boundaries within the system; b. Implements subnetworks for publicly accessible system components that are [Selection: physically; logically] separated from internal organizational networks; and c. Connects to external networks or information systems only through managed interfaces consisting of boundary protection devices arranged in accordance with an organizational security architecture.	The findings related to OpenShift pertain primarily to “key internal boundaries within a system” for this control requirement. For external boundaries to the system, it is recommended to deploy OpenShift within an external authorization boundary controlled by traditional network security approaches. For internal boundaries that may be defined by the customer for separation of workloads, OpenShift uses the OpenShift SDN to provide a unified cluster network that enables communication between containers across the cluster. The level of control enabled for communication depends on the chosen SDN plug-in and the use case for which the customer may choose it. The ovs-subnet plug-in provides a “flat” pod network where every pod can communicate with every other pod and service. The ovs-multitenant plug-in provides project level isolation for pods and services to prevent communication between pods from differing projects. The ovs-networkpolicy plug-in is the most flexible SDN option, providing extremely fine-grained control capability.
SC-8	TRANSMISSION CONFIDENTIALI	The information system protects the [Selection (one or more):	Internet Protocol Security (IPSec) can be enabled to protect traffic in an OpenShift

ID	CONTROL TITLE	CONTROL DESCRIPTION	APPLICABILITY OF OPENSHIFT TO CONTROL
	TY AND INTEGRITY	confidentiality; integrity] of transmitted information.	cluster by encrypting the SDN traffic in the environment. Communication between the OpenShift Master and Nodes can be configured for TLS 1.2 encryption if required. OpenShift out of the box provides containerized stateless HAProxy as a default router for the whole container ecosystem. OpenShift provides ways to perform TLS termination with secure routes: edge, re-encryption, and passthrough. With passthrough route, TLS termination is handled at the pod level.
SC-28	PROTECTION OF INFORMATION AT REST	The information system protects the [Selection (one or more): confidentiality; integrity] of [Assignment: organization-defined information at rest].	Data at the datastore layer can be configured to be encrypted. This allows for secrets to be encrypted when at rest in etcd. Third-party integrations are also available to provide encryption capability for data at rest.
SC-39	PROCESS ISOLATION	The information system maintains a separate execution domain for each executing process.	OpenShift makes use of underlying features of the OS to support process isolation.
SI-16	MEMORY PROTECTION	The information system implements [Assignment: organization-defined security safeguards] to protect its memory from unauthorized code execution.	OpenShift makes use of underlying features of the OS to support memory protection.

Table 1 - FISMA Moderate Applicability Detail to OpenShift

CONCLUSION

OpenShift hosted on Red Hat Enterprise Linux, as reviewed by Coalfire, can be effective in providing support for the outlined objectives and requirements of NIST 800-53 Rev 4 Moderate baseline in support of a FISMA compliance program. Through proper implementation and integration into the organization's greater technical infrastructure and information security management systems, OpenShift may be useable in a FISMA Moderate-controlled environment. Care should be given for the implementation of OpenShift and the use of the platform for the deployment of containers in support of microservice architectures. The organization wishing to use OpenShift should consider the guidance provided by NIST SP 800-190 when designing their implementation.

Coalfire's opinion is based on observations and analysis of the provided documentation, interviews with Red Hat personnel, and hands-on engagement with a lab environment. The provided conclusions are based upon several underlying presumptions and caveats, including adherence to vendor best practices and hardening of configuration as supported by the system components. This solution should be implemented in alignment with the organization's mission, values, business objectives, general approach to security and security planning, and with respect to the overall organizational security and compliance program.

ADDITIONAL INFORMATION, RESOURCES, AND REFERENCES

https://docs.openshift.com/container-platform/3.11/welcome/index.html

https://docs.openshift.com/container-platform/3.11/admin_guide/ipsec.html

https://docs.openshift.com/container-platform/3.11/architecture/networking/routes.html

https://developers.redhat.com/blog/2017/01/24/end-to-end-encryption-with-openshift-part-1-two-way-ssl/

https://docs.openshift.com/container-platform/3.11/install/prerequisites.html

https://docs.openshift.com/container-platform/3.11/admin_guide/managing_networking.html

https://docs.openshift.com/container-platform/3.11/admin_guide/encrypting_data.html

https://docs.openshift.com/container-platform/3.11/security/monitoring.html

https://docs.openshift.com/container-platform/3.11/install_config/master_node_configuration.html#master-node-config-audit-config

https://docs.openshift.com/container-platform/3.11/install_config/master_node_configuration.html#master-node-config-advanced-audit

https://docs.openshift.com/container-platform/3.11/dev_guide/authorization.html

https://docs.openshift.com/container-platform/3.11/admin_guide/manage_rbac.html

https://docs.openshift.com/container-platform/3.11/admin_guide/manage_users.html

https://docs.openshift.com/container-platform/3.11/architecture/core_concepts/projects_and_users.html

https://docs.openshift.com/container-platform/3.11/admin_guide/manage_scc.html

https://docs.openshift.com/container-platform/3.11/architecture/index.html#architecture-index

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html-single/selinux_users_and_administrators_guide/

https://docs.openshift.com/container-platform/3.11/admin_guide/seccomp.html

https://github.com/kubernetes/kubernetes/blob/release-1.4/docs/design/seccomp.md

https://docs.openshift.com/container-platform/3.11/admin_guide/sysctls.html

https://docs.openshift.com/container-platform/3.11/admin_guide/ipsec.html

https://developers.redhat.com/blog/2017/01/24/end-to-end-encryption-with-openshift-part-1-two-way-ssl/

https://docs.openshift.com/container-platform/3.11/architecture/networking/networking.html

https://docs.openshift.com/container-platform/3.11/security/network_security.html

https://docs.openshift.com/container-platform/3.11/install_config/router/index.html

https://docs.openshift.com/container-platform/3.11/install_config/configuring_sdn.html

https://docs.openshift.com/container-platform/3.11/security/container_content.html

https://csrc.nist.gov/projects/risk-management/detailed-overview

https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-190.pdf

https://csrc.nist.gov/Projects/Risk-Management/Risk-Management-Framework-Quick-Start-Guides/Step-0-Prepare

https://csrc.nist.gov/Projects/Risk-Management/Risk-Management-Framework-Quick-Start-Guides/Step-1-Categorize

https://csrc.nist.gov/Projects/Risk-Management/Risk-Management-Framework-Quick-Start-Guides/Step-2-Select

https://csrc.nist.gov/Projects/Risk-Management/Risk-Management-Framework-Quick-Start-Guides/Risk-Management-Framework-(RMF)-Step-6-Monitor

https://csrc.nist.gov/Projects/Risk-Management/Risk-Management-Framework-(RMF)-Overview/Security-Controls

https://csrc.nist.gov/publications/detail/fips/199/final

https://csrc.nist.gov/publications/detail/fips/200/final

https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/final

https://csrc.nist.gov/publications/detail/sp/800-37/rev-1/final

https://www.coalfire.com/Solutions/Audit-and-Assessment/FISMA-Validation

ABOUT THE AUTHORS

Jason Macallister | Contributor | Senior Consultant, Cyber Engineering, Coalfire
Mr. Macallister consults on information security and regulatory compliance topics as they relate to advanced infrastructure and emerging products and solutions.

Mitch Ross | Contributor | Director, Cyber Engineering, Coalfire
As Director, Mr. Ross contributes as an author and thought leader on information security and regulatory compliance topics for Coalfire's clientele with an emphasis in security in the cloud.

Brian Justice | Contributor | Senior Consultant, Cyber Risk Services, Coalfire
Mr. Justice consults on information security and regulatory compliance topics, with an emphasis on U.S. Government frameworks, such as FISMA and FedRAMP.

Published October 2018

ABOUT COALFIRE

Coalfire is the cybersecurity advisor that helps private and public-sector organizations avert threats, close gaps, and effectively manage risk. By providing independent and tailored advice, assessments, technical testing, and cyber engineering services, we help clients develop scalable programs that improve their security posture, achieve their business objectives, and fuel their continued success. Coalfire has been a cybersecurity thought leader for more than 16 years and has offices throughout the United States and Europe. Coalfire.com