WHITE PAPER

RED HAT PRODUCT APPLICABILITY GUIDE FOR PCI DSS V3.2

HOW RED HAT ENTERPRISE LINUX SERVER, RED HAT INSIGHTS, RED HAT ANSIBLE ENGINE, RED HAT ANSIBLE TOWER, AND RED HAT SATELLITE SUPPORT EFFICIENCY FOR A PROGRAM OF COMPLIANCE

JASON MACALLISTER
CHRIS KRUEGER | CISSP, PCI-QSA
FRED KING | CISA, PCI-QSA

VERSION 1.0

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 enterprise grade software products and services built from open source software components using an affordable, predictable subscription and support model. Red Hat engaged Coalfire, a respected Payment Card Industry Qualified Security Assessor (PCI QSA) company, to conduct an independent technical assessment of Red Hat Satellite, Red Hat Insights, Red Hat Ansible Engine, Red Hat Ansible Tower, and Red Hat Enterprise Linux Server. The primary purpose of this product applicability guide is to identify alignment of technical control capabilities of the Red Hat products to the Payment Card Industry Data Security Standard (PCI DSS) v3.2 technical requirements.

Coalfire's reviewed these products individually and as a combined solution with respect to how they may be useful in supporting a payment entity's PCI DSS v3.2 compliance initiatives. Beyond requirement applicability alignment, Coalfire also examined how the aggregated solution may be useful to address challenges for maintaining a program of compliance through centralized system lifecycle management, operational and security insights, and automated remediation. These capabilities can support improved configuration consistency in alignment with compliance requirements.

This product applicability guide may be useful to a payment entity looking for ways to improve management, insights, and control over infrastructure and application life cycles in support of PCI DSS v3.2 compliance. Further, this paper may also be useful to understand how these enterprise tools can be used to support a PCI DSS v3.2 program of compliance. This paper identifies and discusses PCI DSS v3.2 requirements that may be addressed by both the individual and combined capabilities of Red Hat Enterprise Linux Server, Red Hat Satellite, Red Hat Insights, and Red Hat Ansible Engine and Red Hat Ansible Tower.

COALFIRE OPINION

Coalfire determined that Red Hat Enterprise Linux Server can be an effective platform for use in PCI DSS v3.2 assessed environments. Red Hat Enterprise Linux Server comes integrated with many security features and functions that effectively support numerous PCI DSS technical control requirements.

Alongside Red Hat Enterprise Linux Server, Red Hat Satellite, Red Hat Ansible Engine, and Red Hat Ansible Tower can be powerful tools to assist organizations with centralized management and control of systems and applications throughout their lifecycle. The lifecycle management capacity of these tools can assist organizations with maintaining a state of continuous compliance and demonstration of the organization's business as usual technical compliance activities. Many routine tasks carried out on systems, from provisioning through remediation, can be automated to increase efficiency and further enable system-wide compliance consistency.

Finally, Red Hat Insights is a valuable service for providing visibility into the increasingly complex IT environments, enabling security and operations teams to proactively identify and respond to issues that can impact performance, availability, security, and compliance. Likewise, system health and configuration monitoring through Red Hat Satellite can help to ensure that systems are in alignment with organizational standards and are compliant with PCI DSS 3.2 technical requirements.

INTRODUCING PCI DSS V3.2

The Payment Card Industry Security Standards Council (PCI SSC) was founded in 2006 by American Express, Discover, JCB International, Mastercard, and Visa Inc. to serve those who work with and are associated with payment cards. The mission of the PCI SSC is twofold; first, to help merchants and financial institutions understand and implement standards for security policies, technologies, and ongoing processes that protect their payment systems from breaches and theft of cardholder data (CHD); second, to help vendors understand and implement standards for creating secure payment solutions. PCI DSS v3.2 is a framework that defines baseline physical, technical, and operational security controls, defined as requirements and sub-requirements, necessary for protecting payment card account data. PCI DSS defines two categories of payment card account data: CHD, which includes primary account number (PAN), cardholder name, expiration date, and service code; and sensitive authentication data (SAD), which includes full track data (magnetic stripe or equivalent on a chip), card security code (CAV2/CVVC2/CVV2/CID), and personal identification numbers (PINs/PIN blocks) entered during the transaction.

UNDERSTANDING PCI DSS PURPOSE AND SCOPE

A reliable approach for determining where PCI DSS is required to be applied begins with identification and definition of scope for review. As stated on page 5 of the PCI DSS Requirements and Security Assessment Procedures, Version 3.2, April 2016, PCI DSS applies to any organization that stores, processes, or transmits CHD. These organizations include, but are not limited to: merchants, payment processors, issuers, acquirers, and service providers. The PCI DSS security requirements apply to all system components included in or connected to the cardholder data environment (CDE). The CDE is comprised of people, processes, and technologies that store, process, or transmit CHD or SAD (PCI Security Standards Council, 2016). PCI DSS defines twelve requirements designed to address six objectives, as show in the high-level overview displayed in Table 1.

OBJECTIVES	REQUIREMENTS
Build and Maintain a Secure Network and Systems	1. Install and maintain a firewall configuration to protect cardholder data 2. Do not use vendor-supplied defaults for system passwords and other security parameters
Protect Cardholder Data	3. Protect stored cardholder data 4. Encrypt transmission of cardholder data across open, public networks
Maintain a Vulnerability Management Program	5. Protect all systems against malware and regularly update anti-virus software or programs 6. Develop and maintain secure systems and applications
Implement Strong Access Control Measures	7. Restrict access to cardholder data by business need to know 8. Identify and authenticate access to system components 9. Restrict physical access to cardholder data
Regularly Monitor and Test Networks	10. Track and monitor all access to network resources and cardholder data 11. Regularly test security systems and processes
Maintain an Information Security Policy	12. Maintain a policy that addresses information security for all personnel

Table 1 - PCI DSS - High-Level Overview

SCOPING AND SEGMENTATION

PCI DSS recommends that an assessed entity confirm the accuracy of their PCI DSS scope at least annually or prior to the annual assessment. To help identify scope, the payment entity should evaluate their systems to identify all locations and flows of CHD and identify all systems that are connected to or, if compromised, could impact the CDE. Systems that store, process, or transmit CHD/SAD are categorized as CDE systems. Likewise, system components that reside on the same network segment (for example, same subnet or VLAN) as systems that store, process, or transmit CHD/SAD are considered "CDE Systems". A second category in scope for assessment is "Connected-to or Security-impacting Systems". A third category of systems as defined by PCI DSS is called "Out-of-Scope Systems". Out-of-scope systems, by definition, are typically not considered in scope for assessment; however, it is important to understand that any system that is part of a payment entity's infrastructure can potentially, through vulnerability or lack of adequate security control, provide a foothold to a zone of the payment entity's internal network, which may be used to, over time, pivot and acquire increasing access until CHD is eventually compromised (PCI Security Standards Council, 2016).

PCI DSS recommends the use of network segmentation to isolate the CDE from the remainder of an entity's network as a method to reduce the scope of PCI DSS assessment, the cost of a PCI DSS assessment, the cost and difficulty of implementing and maintaining PCI DSS controls, and the risk to an organization by consolidating CHD into fewer, more controlled locations. The intent of segmentation is to prevent traditionally out-of-scope systems from being able to communicate with systems in the CDE and/or impact the security of the CDE (PCI Security Standards Council, 2016).

A PROGRAM OF CONTINUOUS COMPLIANCE

PCI DSS specifies a best practice for implementing PCI DSS into Business-as-Usual (BAU) processes. These best practices are designed to ensure that prescribed security controls continue to be implemented properly, to monitor the effectiveness of security controls on an ongoing basis, and to maintain an entity's PCI DSS compliant environment in between PCI DSS assessments. The recommended best practices for implementing BAU processes include monitoring of security controls, ensuring that failures in security controls are detected and responded to in a timely manner, and reviewing changes to the environment that may impact PCI DSS scope, changes to organizational structure, and the performance of periodic reviews (PCI Security Standards Council, 2016).

SUGGESTIONS FOR USE OF THIS REPORT

This white paper is intended to be used by a variety of payment card entities and other interested parties that may be involved in the sale, construction, operation, or assessment of infrastructure using Red Hat Enterprise Linux, Red Hat Satellite, Red Hat Insights, Red Hat Ansible Engine, and/or Red Hat Ansible Tower. It guides Red Hat customers with understanding how these solutions can be used to support a program of compliance. It provides examples of how this assessment may be utilized by the identified entities engaged in a PCI DSS lifecycle including merchants and financial institutions; payment solution service providers; Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) service providers; designated entities and others who share responsibility with a payment entity; and PCI DSS QSAs.

INTRODUCING RED HAT SOLUTIONS

Red Hat is the world's leading provider of open source solutions, using a community-powered approach to provide reliable and high-performing cloud, virtualization, storage, Linux, and middleware technologies. The transparency of Red Hat's open source community-powered approach enables a greater scope of design diversity than alternative closed development approaches. Due to this transparency, open source solutions have the propensity to be more inherently secure. A larger community of developers and users are able to identify and correct discovered flaws on a more timely basis. Beyond this collaborative approach to delivering an enterprise operating platform and tools, Red Hat offers support, training, and consultation to help their customers successfully utilize the solutions that Red Hat provides.

ENTERPRISE PLATFORM FOR THE MODERN DATA CENTER

A platform for a modern data center should be able to keep up with the diverse demands of entities that plan to use it. It should be adaptable, flexible, and sustainable to support ongoing hardware innovations. It should be able to support a diverse set of applications and business solutions with a wide array of delivery options efficiently, effectively, reliably, and securely across private, public, and hybrid clouds. As an enterprise solution, it should be a consistent and stable platform. It should also be backed by support, tools, and solutions to optimize IT using the entity's existing resources. Ideally, the platform should be designed with security at the inception.

The security built into the platform should be adaptable to address the ever-changing and constantly evolving threat landscape. Simply protecting the entity's boundary between "trusted" internal networks and "untrusted" or "unmanaged" external networks is no longer sufficient. Attackers are increasingly finding ways to gain access to the internal network, often through insider access, and once inside can often pivot undetected across the "trusted" network. Placement of security controls closer to the sensitive or critical data is a way to reduce this risk. A zero-trust architecture, through well planned and executed access controls (least access), network controls, eliminating unnecessary services (least function), use of cryptography, and other security techniques, is useful to minimize the surface area of attack for internal assets and make it more difficult for attackers to find and gain access to protected data. The concept of a zero-trust architecture assumes that a breach has already occurred and uses controls to minimize the impact of that breach. Red Hat Enterprise Linux Server includes hardening guidance, as well as several security control features that are useful to support a zero-trust architecture.

Red Hat Enterprise Linux Server

The foundation of Red Hat's offering is Red Hat Enterprise Linux Server. There is broad applicability for Red Hat Enterprise Linux Server as a platform to serve various business applications and components including, but not limited to, traditional bare metal servers, hypervisors, virtual machines, and container platforms. The discussion of Red Hat Enterprise Linux Server for this paper is broad, but the security features, functionality, and capabilities are applicable across many of the use cases. For the purpose of this paper, Coalfire considers the use of Red Hat Enterprise Linux Server as a possible component of a payment entity's CDE, which may have a role in storing, processing, or transmitting CHD/SAD, and/or a supporting system.

Red Hat Enterprise Linux Server is a platform capable of supporting a multitude of application architectures. It provides fundamental core operating system functions including hardware resource management capabilities to orchestrate an infrastructure's basic computing requirements including CPU, memory, network, and storage. Beyond the traditional operating system basics, Red Hat has incorporated a number of security features and functions that are useful for entities looking to improve security or support various compliance initiatives.

Red Hat provides guidance for securing Red Hat Enterprise Linux Server, which can be useful for security and operations personnel planning for secure deployments of the operating platform. These hardening guides are broken out by topics that align with general security best practices and common requirements found within various compliance frameworks. The guidance provided by Red Hat for Red Hat Enterprise Linux Server is applicable and useful in general to varying degrees for aligning with PCI DSS requirements 1, 2, 3, 6, 7, 8, 10, and 11. Detailed alignment can be found later in this document.

In addition to hardening best practices, Red Hat has incorporated a number of available security features and functions into Red Hat Enterprise Linux Server. These features, when implemented, can help to improve the management and functional security of the server platform across various deployment modes. Some of the integrated security features and functions that may be useful in a PCI DSS regulated environment include disk and file level encryption, encryption for data in transit, Advanced Intrusion Detection Environment (AIDE), Domain Name System Security Extension (DNSSEC), USBGuard, Media Access Control Security (MACsec), Security Enhanced Linux (SELinux), identity management (IdM), compliance and vulnerability scanning with Open Security Content Automation Protocol (OpenSCAP), system auditing, firewalld, and iptables.

Data-At-Rest Disk Encryption

Red Hat Enterprise Linux Server provides a number of data encryption options either for data at rest or data in transit. Many of the requirements in PCI DSS v3.2 are applicable to how the payment application and/or payment hardware handle encryption and how the payment entity generates and secures encryption keys for the protection of CHD and SAD. While the encryption provided by Red Hat Enterprise Linux Server does not specifically address all of the requirements and methodologies as it pertains to secure payment application design, it can be useful for providing additional layers of security for specific use cases. Red Hat Enterprise Linux Server complies with the Federal Information Processing Standard (FIPS) Publication 140-2 and can be made compliant with the standard by enabling FIPS mode on the server.

For data at rest, Red Hat Enterprise Linux Server includes Linux Unified Key Setup-on-disk-format (LUKS) disk encryption for full disk or partition encryption. This is useful for securing the contained data on disk when the server is powered off, when the disk is removed, or for removable media. Multiple ciphers, cipher modes, and initial vectors are supported for enabling at-rest encryption on the server. Red Hat Enterprise Linux Server also offers Policy Based Decryption (PBD) one of the implementations of which is called Network Bound Disk Encryption (NBDE). NBDE allows for entire disks to be encrypted, including the boot disk. For NBDE encryption, the target server enrolls with a remote server where the keys for encryption and decryption are stored. The server involved with managing the keys and encryption for servers are the network is needed in order for the disks to be decrypted. The entity can set up NBDE such that disks are unaccessible or unbootable and incapable of being decrypted if removed from the facility or from the system on the network segment where NBDE is being used. The following steps outline the process for encryption or decryption of disks using NBDE:

1. Phase 1: Enrollment
    a. During this phase the encryption key for the LUKS volume is encrypted with the public key of the remote decryption server.
        i. This operation can occur without direct access of the server.
2. Phase 2: Decryption
    a. The client generates a random key and wraps already encrypted volume key with a special cryptographic operation. Then it sends the results to the server
    b. The server decrypts the payload by removing the PKI encryption but the content is still encrypted with the wrapping key so the server does not see the actual data.
    c. The client receives the response from the server and decrypts the payload to get direct access to the key.
    d. The client then uses the key to unlock and mount the volume.

On the client side is the Clevis framework. On the server side, on the network, is the system that will do the remote unlocking that utilizes a TANG service. Figure 1 illustrates the components of NBDE for network-based encryption and decryption of disks.

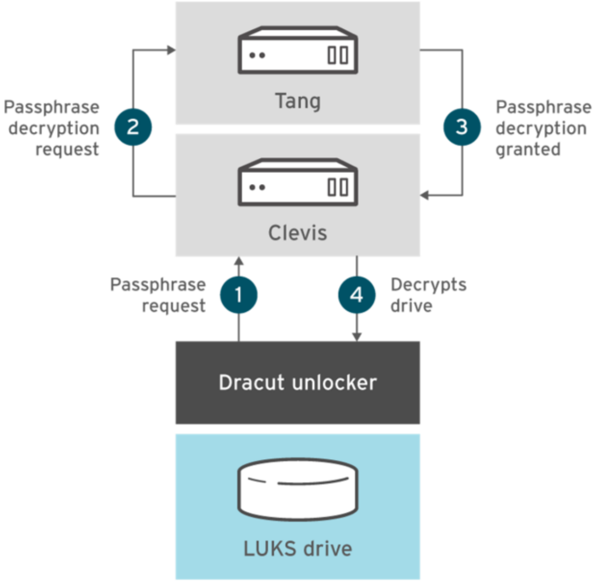


Figure 1: NBDE Encryption Components

Data-in-Transit Encryption

For transmitted data, Red Hat Enterprise Linux Server provides options for securing communication either using secure virtual private networks (VPNS), secure shell (SSH) with OpenSSH, OpenSSL, and stunnel. In Red Hat Enterprise Linux Server, VPNS can be configured using the IPsec tunneling protocol for host-to-host and/or site-to-site. SSH with OpenSSH provides a secure connectivity tool for remote login with the SSH protocol. OpenSSH encrypts traffic to protect against eavesdropping, connection hijacking, and other attacks. OpenSSL is a library that provides cryptographic protocols to applications. The stunnel program is an encryption wrapper between a client and a server. It listens on the port specified in its configuration file, encrypts the communication with the client, and forwards the data to the original daemon listing on its usual port. This way, any service can be secured where the service itself does not support any type of encryption. Likewise, it can be used to improve the security of a service that uses a type of encryption that is not strong enough for the specific application, for instance SSL version 3 affected by the POODLE SSL vulnerability (SSL 2.0 is intentionally not supported on RHEL 7.4 and later).

Red Hat Enterprise Linux Server provides support for the latest version of TLS. Included for support of the TLS protocol is OpenSSL, GnuTLS, and NSS toolkits. Additional guidance is provided by Red Hat for configuration of specific applications for TLS support such as Apache HTTP Server. Various cipher suites are also available for TLS-secured communications.

For additional secure authentication mechanisms and multi-factor authentication support, Red Hat Enterprise Linux Server also provides Smart Card and Hardware Security Module (HSM) integration.

MACsec

MACsec encrypts and authenticates all traffic in LANs with the GCM-AES-128 algorithm. MACsec can protect not only IP, but also Address Resolution Protocols (ARP), Neighbor Discovery (ND), or DHCP. While IPsec operates on the network (layer 3) and SSL and TLS on the transport layer (layer 4), MACsec operates in the data link layer (layer 2). Because it operates on layer 2 protocol, it can secure traffic from higher layer protocols as well. MACsec is an extension to 802.1x and provides secure key exchange and mutual authentication for MACsec nodes.

AIDE

AIDE is a utility on Red Hat Enterprise Linux Server that creates a database of files on the system and then uses that database to ensure file integrity and detect system intrusions. AIDE can be scheduled to run during regular intervals such as weekly at a minimum or daily at most.

DNSSEC

DNSSEC enables a DNS client to authenticate and check the integrity of responses from a DNS nameserver in order to verify their origin and to identify if they have been tampered with in transit. Traditionally, DNS lookups are done insecurely and susceptible to man-in-the-middle attacks due to lack of authentication. Using insecure DNS lookups means that the client cannot have confidence that the replies that appear to have come from a given DNS name server are authentic and have not been tampered with. In the same way, a recursive name server cannot be sure that records that it obtains from other nameservers are genuine. An attacker can intercept traditional DNS requests and reply with false information. In this way, an attacker can redirect a client to a website or service that he or she controls.

USBGuard

The USBGuard software framework provides systems protection against intrusive USB devices by implementing whitelisting and blacklisting capabilities based on device attributes. To enforce a host-based policy, USBGuard uses a Linux kernel USB device authorization feature. This feature is typically more important on end user devices such as workstation and laptops. Servers in the data center should be physically protected to prevent unauthorized users from attaching USB devices. However, as an added measure to protect against insider threats, USBGuard can be implemented on servers.

SELinux

Red Hat Enterprise Linux Server comes with the SELinux Linux Kernel security module enabled by default. SELinux is an implementation of mandatory access control (MAC) mechanisms in the Linux kernel and provides an additional layer of access control. After discretionary access controls (DAC) are checked, SELinux checks to determine if the requested operation is allowed. Flaws in programs or services that run with coarse-grained privileges can potentially be exploited to escalate further systems access. SELinux provides finer-grained control over the activities that programs and services can take, even when executed by privileged users. Above identity and ownership access decisions, SELinux verifies the role of the user, the function and trustworthiness of the program, and the sensitivity and integrity of the data to determine if the action can be authorized.

IdM

Red Hat Enterprise Linux Server provides a centralized way to manage identities and define access-control policies for users, machines, and services within large Linux and UNIX enterprise environments. Red Hat Enterprise Linux IdM is a way to create identity stores, centralized authentication, domain control for Kerberos and DNS services, and authorization policies, all on Linux systems using native Linux tools. It provides a unifying skin for standards-defined, common network services, including PAM, LDAP, Kerberos, DNS, NTP, and certificate services, allowing Red Hat Enterprise Linux Server to serve as domain controllers. IdM domain controllers can deliver enterprise-level single-sign-on, certificate management, DNS integration, and command-line and web user interfaces (UI) for managing enterprise identities, certificates, and keys.

IdM is a management tool for Linux domains. It centralizes identity management and identity policies; whereas without IdM, each Linux server must be individually managed with unique identities and policies per server. IdM is built on existing native Linux applications and protocols. The underlying technologies of IdM would be more familiar for Linux administrators. The goal of IdM is to simplify administrative overhead by allowing users, machines, services, and policies for Linux systems to all be configured in one central place, using the same tools. Figure 2 graphically depicts the elements of IdM that can be administered for Linux and Unix systems in the IdM domain.

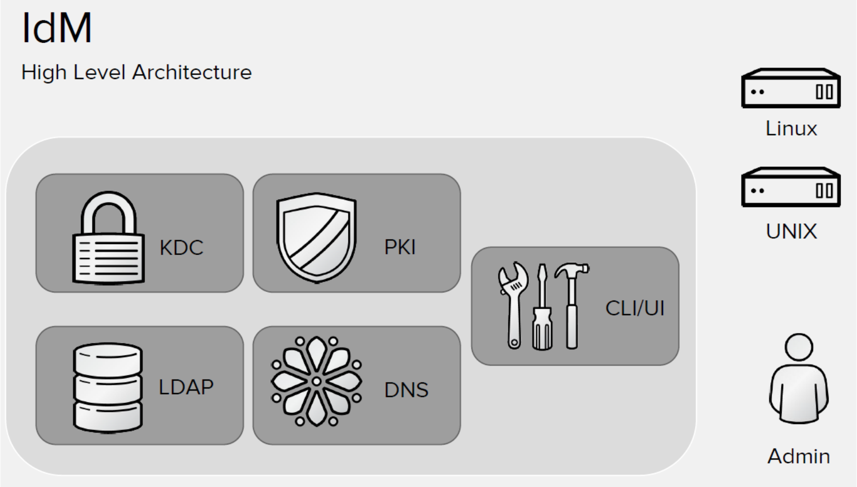


Figure 2: IdM High-Level Architecture

Red Hat Enterprise Linux IdM provides more advanced capabilities than LDAP, with support for advanced security policies like sudo, host-based access control rules, automount, netgroups, SELinux user mappings, and other similar capabilities. In many environments, it is not uncommon for user accounts for both Linux and Windows to be stored in Windows Active Directory. Red Hat Enterprise Linux IdM can coexist with Active Directory to provide additional identity management services for Linux systems. IdM simplifies maintenance of multiple domains by supporting interoperability with Microsoft Active Directory. For interoperability, Red Hat Enterprise Linux IdM provides two options: direct integration where Red Hat Enterprise Linux systems are joined directly into an Active Directory domain and indirect access through cross-realm Kerberos trusts between IdM in Red Hat Enterprise Linux Server and an Active Directory forest.

Red Hat recommends an indirect integration approach, preferably using a trust-based solution leveraging IdM in Red Hat Enterprise Linux Server as the central server to control Linux systems and then establishing a cross-forest Kerberos trust with Active Directory. At the most basic level, Red Hat Enterprise Linux IdM is a domain controller for Linux and Unix machines. This approach is recommended due to the complexity that can be associated with a growing number of Linux systems in an entity’s environment. As the number of Linux systems grows, there is a greater need for centralized management of the identity-related policies that are unique to Linux systems such as host-based access control, sudo, or SELinux user mappings. Without IdM, these identity policies are commonly found in local configuration files unique to each system. To help maintain consistency and accuracy across Linux servers in a domain, it is important to be able to centrally manage these policies as they are distributed across the domain. Figure 3 depicts the relationship between Linux systems, IdM, and Active Directory. This design allows for authentication of Active Directory users to Linux systems indirectly through IdM.

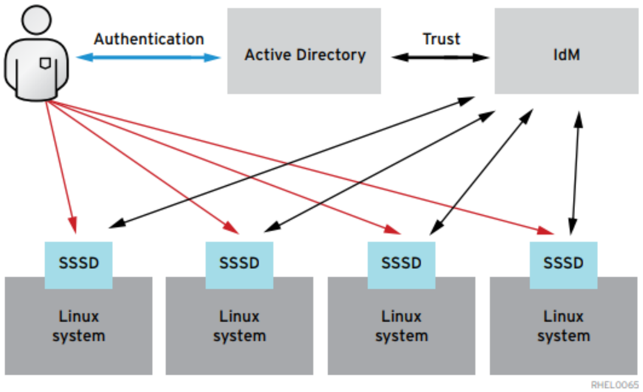


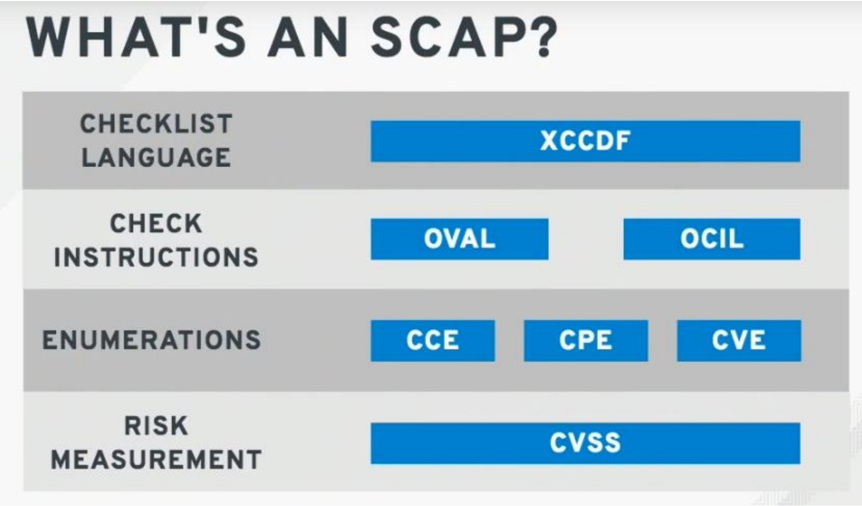
Figure 3: Cross-Realm Trust between Active Directory and Red Hat Enterprise Linux IdM

Communication between the Linux system and the IdM is brokered with the system security services daemon (SSSD) also shown in Figure 3.

OpenSCAP

The Security Content Automation Protocol (SCAP) is a U.S. standard maintained by the National Institute of Standards and Technology (NIST). The OpenSCAP project is a collection of open source tools for implementing and enforcing this standard. Included into Red Hat Enterprise Linux Server, OpenSCAP is an auditing tool that creates a standard security checklist for enterprise systems. It provides practical security hardening advice for Red Hat Enterprise Linux Server and other Red Hat products and links this advice to compliance requirements, making deployment activities like certification and accreditation easier. SCAP provides machine-readable controls that can automate compliance checks. Incorporated into Red Hat Enterprise Linux Server, this allows administrators to verify or certify technical compliance with various compliance frameworks, including PCI DSS v3.2. It can be the basis for security automation strategies for implementation and management of system standards. This also can allow administrators to determine if there are misconfigurations that may negatively impact compliance and respond to remediate compliance drift.

The compliance baselines for OpenSCAP are flexible to allow for tailoring to additional entity-specified standards. Figure 4 breaks out the components of an SCAP from top down. The Extensible Configuration Checklist Description Format (XCCDF) is a standard way of expressing checklist content and defines security checklists. It is made up of check instructions from the Open Vulnerability and Assessment Language (OVAL) and the Open Checklist Interactive Language (OCIL). OVAL is an information security community effort to standardize how to assess and report on the machine state of computer systems. OCIL defines a framework for expressing a set of questions to be presented to a user and corresponding procedures to interpret responses to these questions. OCIL is typically used for compliance checks that are more difficult to apply digitally, e.g., physical security around a particular system. These checklists are combined with other specifications such as Common Platform Enumeration (CPE), Common Configuration Enumeration (CCE), and Common Vulnerabilities and Exposures (CVE) to create a SCAP-expressed checklist that can be processed by SCAP-validated products. Additionally, OpenSCAP verifies the presence of patches using content produced by the Red Hat Security Response Team (SRT), checks system security configuration settings, and examines systems for signs of compromise by using rules based on standards/specifications. The risk measurement for OpenSCAP is based on the Common Vulnerability Scoring System (CVSS). This allows a numerical representation reflecting the severity of a discovered vulnerability as a result of a missing patch or misconfiguration. The numerical representation helps security operations to focus on higher priority issues that present the greatest risk from a top down approach.



WHAT'S AN SCAP?

CHECKLIST LANGUAGE	XCCDF		
CHECK INSTRUCTIONS	OVAL	OCIL	
ENUMERATIONS	CCE	CPE	CVE
RISK MEASUREMENT	CVSS		

Figure 4 - SCAP Components

When combined with a tool to centralize and automate auditing such as Red Hat Satellite, this can be a powerful way for entities to understand and report on compliance state on a continual basis. Figure 5 illustrates the process for evaluating systems. SCAP security guides, including policy guidance for PCI DSS, is input into OpenSCAP, systems are evaluated against these policy standards, and standardized results are output to provide a snapshot of compliance state.

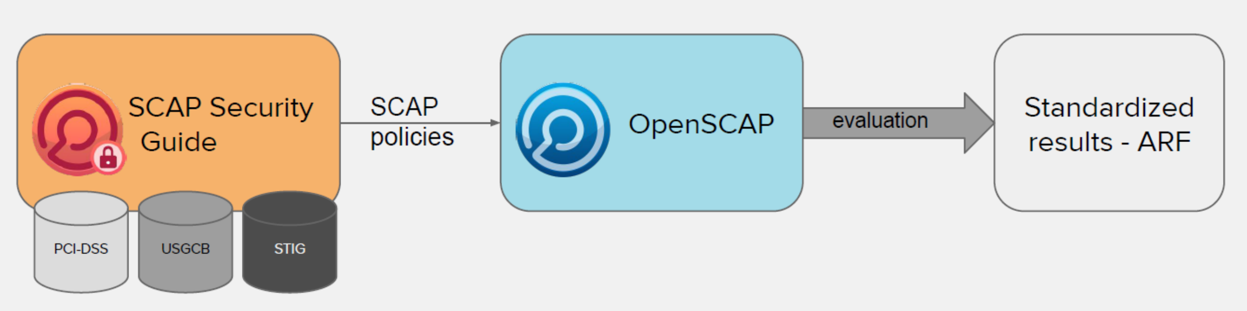


Figure 5: OpenSCAP Evaluation Process

Red Hat Enterprise Linux Server 7.5 includes improved support for delivery of compliance and security configurations at scale through the integration with OpenSCAP in Red Hat Enterprise Linux Server with Red Hat Ansible Automation (Red Hat Ansible Engine and Red Hat Ansible Tower), Ansible playbooks can be generated directly from scans. The resulting playbooks can be used by Red Hat Ansible Automation to implement remediation rapidly and consistently across the enterprise.

firewalld

firewalld provides a dynamically managed firewall on the Red Hat Enterprise Linux Server host with support for network/firewall zones that define the trust level of network connections or interfaces. firewalld supports a separation of runtime and permanent configuration. One advantage of firewalld is that firewall configuration or policy changes are immediately effective at runtime, whereas for iptables, changes to configuration require a complete reload of the firewall configuration. In other words, with firewalld, no restart of the service or daemon is needed for the firewall policy to go into effect. Separation of runtime and permanent configuration allows for evaluation and tests in runtime prior to making a change permanent.

Once the runtime configuration has been tested, the configuration can be saved to the permanent environment so that it is automatically applied the next time the system is restarted.

firewalld can be used to separate networks into different zones based on the level of trust the user has decided to place on the interfaces and traffic within that network. firewalld comes with a preset number of zones that are common to most environments; however, default zone names and settings are proposals that can be changed according to the needs of the entity. For instance, administrators can designate a CDE zone for CDEs that are designated separately from the non-CDE zones. This can be helpful for making determinations of data flow based on policy sets that either permit, deny, or drop based on the entity's requirements and what is defined in their security standard.

MANAGEMENT TOOLS TO SUPPORT AND MAINTAIN A PROGRAM OF COMPLIANCE

To increase visibility and management efficiency for the IT operations around Red Hat Enterprise Linux Server, Red Hat provides a set of management tools. These tools include Red Hat Insights, Red Hat Satellite, and Red Hat Ansible Tower and Red Hat Ansible Engine.

Red Hat Insights

IT operational analytics technologies are primarily used to discover complex patterns in high volumes of noisy system availability and performance data. Modern IT operations are often complex with a combination of technologies being used for business delivery including private cloud, public cloud, virtualization technologies, container technologies, software-defined networks, virtual networking, and so forth. Every component of these complex environments generates data, often producing a large volume of data, whether through audit and event log generation, performance data, wire data, and so on. These infrastructures, in and of themselves, are often complex; furthermore, the data that they generate to understand system state at any given time can be complex and voluminous. Though this information can be made readily available for periodic review, the time and effort to analyze and understand the data can be staggering. Consequently, many IT operations may simply ignore much of the generated data. To make the preponderance of data more manageable, operators may choose to focus their review on a small subset of specifically targeted information based on perceived compliance requirements. For everything else, the option is to respond to problems as they occur with a possibility of adding lessons learned in incident response to future auditable events.

Hidden issues, by way of insufficient network and system visibility, may often remain undetected until they surface as problems. This leaves IT staff to respond to issues after they are discovered as a result of disruptive events that potentially have negative impact to the confidentiationality, integrity, and availability of the entity's business systems. At this point, remediation is typically part of the incident response process; however, some damage may have already been done. These events can ultimately lead to loss in revenue, loss in production, embarrassment, and the possibility of fees and or fines being levied by regulatory agencies. Having greater visibility into the state of systems can equip IT organizations to identify issues or risks and be proactive in remediating these risks or issues before they become incidents.

Red Hat Insights is provided as a software-as-a-service (SaaS)[1] offering from Red Hat. Red Hat Insights supports analysis, remediation, and automated resolution of security, configuration, and performance impacting issues for Red Hat infrastructures. Red Hat Insights uses intelligent data to pinpoint technical risk

[1] As a service provider, the degree to which Red Hat is responsible for PCI DSS compliance is somewhat of a grey area. The type of data that is being sent to Red Hat Insights for analysis should not contain CHD or SAD. However, the extent to which Red Hat Insights, being a CDE supporting system, can impact a CDE's security should be reviewed by the customer. Red Hat provides the means for the service subscriber to identify the data that is being sent and to black list data as the customer determines is necessary. At a minimum, for due diligence, a payment entity may choose to include Red Hat in their listing of service providers with a description of the service provided per Requirement 12.8 and 12.8.1. The payment entity may also desire to maintain a written agreement that includes acknowledgement for responsibility for security to the extent that the service provided could impact the customer's CDE per requirements 12.8.2 and 12.9. Red Hat may choose to disclose to their customer how the security of the service that is provided including communication channels over insecure networks is handled and how the customer's data is protected.

in the organization’s infrastructure and helps IT resolve problems before the business is adversely impacted. This process is called prescriptive analytics. Prescriptive analytics helps to identify what issue is likely to occur, why it is likely to occur, and how to fix the issue to avoid occurrence. It supports operational analytics and automated resolution across physical, virtual, container, and private and public cloud environments.

System metadata is collected in real-time from the infrastructure, fed into Red Hat Insights on a scheduled basis, and analyzed against a growing collection base of common performance, stability, availability, and security issues informed by a process of historical discovery, verification by Red Hat experts, and creation of an automated process to fix it. The basis of this knowledge that feeds Red Hat Insights includes a growing history of identification and resolution for over 1 million solved cases, across 30,000 unique solutions, using over 700 certified engineers globally who are participating with this on an ongoing basis. The rule base is partially based on statistical frequency and is critical issue targeted. When infrastructures are compared against this database of rules, Red Hat Insights can help IT teams to prioritize detected issues and quickly identify where within the infrastructure the problems exist. Red Hat Insights provides tailored remediation steps to fix discovered issues, often before the threat is realized by the end-user communities. The remediation steps can be automated through the integration with Ansible.

Red Hat Insights only collects the minimum amount of data necessary to be able to provide actionable insights. The system metadata that is collected and sent to Red Hat Insights is encrypted both in flight and at rest. There is also an ability for the customer to blacklist data that is sent to Red Hat Insights. If data is chosen for blacklist, it will likely impact the ability to identify actions that may need to be taken against a particular system.

Alerts are called actions in Red Hat Insights because the intelligence provided by Red Hat Insights is intended to be acted upon to correct the discovered issues with tailored remediation for a specifically identified host. Actions in Red Hat Insights are broken out into four categories: availability, stability, performance, and security. Selecting a category will show the the unique actions that have been identified associated with that category along with the number of systems that are impacted by the action. Actions are ranked by severity to help organizations focus on actions with the highest level of risk first. The action rankings include Critical, High, Medium, and Low. Additionally, an individual host within the Red Hat Insights dashboard can be selected to see all its associated actions. The associated actions can automatically generate an Ansible playbook for automated remediation. The proactive analysis is aware of multiple factors that may determine the level of vulnerability that occurs with a particular system. With respect to security, for example, a system may have a known vulnerability as informed by a particular CVE, which requires the system to be patched. In addition, the same system may also be listening on a port that allows for that vulnerability to be remotely exploited. The combination of factors escalates the criticality. The highlighted criticality due to presence of vulnerability and opportunity to exploit can be useful to operations teams for prioritizing resolution of issues within the environment.

Red Hat Insights can be integrated with Red Hat Satellite where Red Hat Enterprise Linux Server is managed. The Red Hat Enterprise Linux Servers communicate with Satellite. Satellite can then broker communication of the relevant data to Red Hat Insights, rather than requiring the host to communicate with Red Hat Insights directly. In this way, Red Hat Satellite acts as a proxy for data sent to Red Hat Insights for further analysis.

Red Hat Satellite

Red Hat Satellite is a system management solution that enables its users to deploy, configure, and maintain their systems across physical, virtual, and cloud environments. Red Hat Satellite provides provisioning, remote management, and monitoring of multiple Red Hat Enterprise Linux Server deployments with a single, centralized tool. Red Hat Satellite Server synchronizes content from the Red Hat Customer Portal and other sources and provides functionality including fine-grained life cycle management, user and group role-based access control, integrated subscription management, and advanced GUI, CLI, or API access.

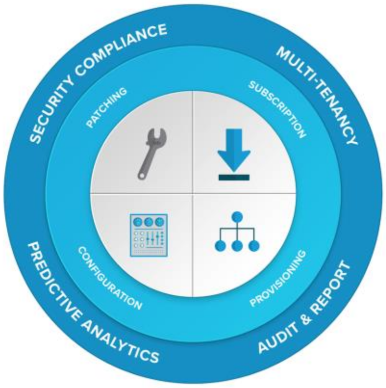


Figure 6: Red Hat Satellite Complete Life Cycle Management

The Red Hat Satellite architecture is depicted in Figure 7. From top down, this illustrates the content sources, both external and internal, for Red Hat Satellite, Red Hat Satellite Server, Red Hat Satellite Capsule Servers, and Red Hat Satellite managed hosts.

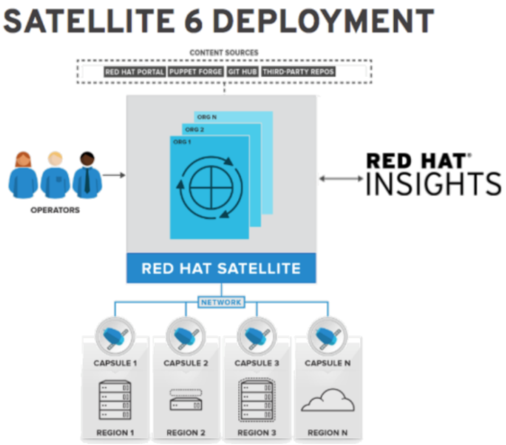


Figure 7: Red Hat Satellite Architecture

The primary purpose of Red Hat Satellite is to make Red Hat infrastructures easier to deploy, scale, and manage. This management tool helps users provision, configure, and update systems to keep them running efficiently, securely, and in compliance with various entity standards. Red Hat Satellite also provides the means to automate regular system administration tasks. Red Hat Satellite provides a central console to give administrators a single location for accessing status reports and for provisioning, configuring, and updating systems. Through Red Hat Satellite, administrators can provision, configure, and update managed hosts on many different platforms including VMware vSphere, Red Hat Enterprise Virtualization, OpenStack, KVM, Amazon EC2, GCE, Rackspace, bare metal, and Docker.

Larger, more complex operations can be difficult to manage. Processes that require management of assets on a host by host basis can be difficult, if not labor intensive. Using a manual unit by unit process, that is unable to keep up with the businesses demands can lead to unpatched or under patched systems, system misconfigurations, and other issues that can impact availability and security. An attacker only needs one vulnerable system in an environment to gain a foothold, escalate privileges, and begin to further exploit other systems on the network. Benefitting a program of compliance, Red Hat Satellite allows organizations to define and establish Standardized Operating Environments (SOE) that can allow the organization to deploy in a fast and repeatable manner. This helps organizations to improve system consistency, which helps to increase efficiency, reduce errors, and meet compliance standards. Rather than manage systems individually, an administrator can manage many assets as though he or she was managing a single asset.

Through a central management console, systems can be automatically configured to meet organizational and compliance standards. Administrators can easily view and report on configuration issues that may negatively impact compliance and address them in a timely manner. Administrators can also see where changes have occurred in the environment and address where configurations have drifted from the defined standards. The same OpenSCAP capabilities which are provided for a single system in Red Hat Enterprise Linux Server are available with Red Hat Satellite at scale to allow for identification of configuration issues that impact compliance. Systems are measured against the compliance standards established through OpenSCAP, and Red Hat Satellite provides a means to identify, report, and automatically remediate issues.

Red Hat Satellite is connected to content provided from the Red Hat Customer Portal, which includes updates, patches, and packages for Red Hat software. This content is made available to Red Hat Satellite to allow organizations to address vulnerabilities resulting from unpatched or under patched systems. With Red Hat Satellite, patching can be scheduled and automated to ensure that vulnerabilities are addressed on a timely basis. Red Hat Satellite includes reporting capabilities to allow entities to identify and address unpatched systems.

Red Hat Satellite takes information from the content sources, analytics provided by Red Hat Insights, technical control requirements outlined with OpenSCAP, and specific organization standards to identify the health of the infrastructure with respect to performance, security, and compliance. Remediation automation from a central console allows for timely response to security and compliance issues. The quick discovery of issues and the automation of remediation can significantly reduce time to resolution over traditional hands-on methods of remediation where administrators interact with each server individually to identify and fix issues.

Red Hat Satellite also provide the means for functional groups to control their own systems. Red Hat Satellite resources are divided into logical groups that can be based on ownership, purpose, content, security level, geographical location, and/or other divisions. Red Hat Satellite, through role-based access control and published content views, can provide specific access to assigned groups for interaction with the assets that the group controls and to the degree that group is granted control capability. The published content views provide a catalog of approved content from available content sources made available for functional groups. Functional groups, for instance, could be broken out by development, test, and production. Likewise, other functional groups may have more specific roles to play within the organizational structure of the entity such as operating platform teams, application support teams, and so forth. This allows for both distribution of control along with granular control capability throughout the entity.

Satellite is also capable of being deployed geographically using Satellite Capsule Servers. Capsule Servers provide content federation and run localized services to discover, provision, control, and configure hosts. The Satellite Capsule Servers can be configured to mirror content from the Satellite Server. The Red Hat Satellite Capsule Server can also be configured to run services for host management such as DHCP, DNS, TFTP, Realm, Puppet Master, Puppet Certificate Authority, Baseboard Management Controller (BMC), provisioning template proxy, and OpenSCAP. Similarly, Red Hat Satellite Capsule Servers can be deployed for use in specific security zones to provide a step of separation or isolation between the sensitive assets. This allows for the benefits of Red Hat Satellite and the supporting services to be employed while maintaining a degree of isolation for protected or regulated workloads.

Red Hat Ansible Tower and Red Hat Ansible Engine

As IT stacks grow and evolve, they become more complex. These complex IT stacks can often generate distinct and purposeful manual processes for management and maintenance of each unique element. Each distinct deployment type may have its own set of disparate manual processes. Over time, it can become difficult for IT administrators to keep up with demand whereby routine manual processes fall behind schedule. Red Hat Ansible Automation allows for repetitive and routine tasks to be automated. Ansible is a simple automation and orchestration engine. It is also a simple human-readable automation language that anyone in the IT organization can understand. No special coding skills are required to use Ansible. Red Hat Ansible Automation is made up of two components: Red Hat Ansible Engine and Red Hat Ansible Tower.

Red Hat Ansible Engine includes the Ansible execution engine, as well as hundreds of modules that enable users to automate all aspects of IT environments and processes. Red Hat Ansible Engine provides for the building and writing of playbooks for enterprises that are typically carried out by development teams. Ansible starts with a user who writes an Ansible playbook. The Ansible playbook invokes Ansible modules in tasks. Tasks are executed sequentially in order from top to bottom. The modules are executable bits of code that get pushed out to target machines. These modules are used to act against an inventory that is made up of hosts and groups. Figure 8 depicts the Ansible Engine architecture at a general level.

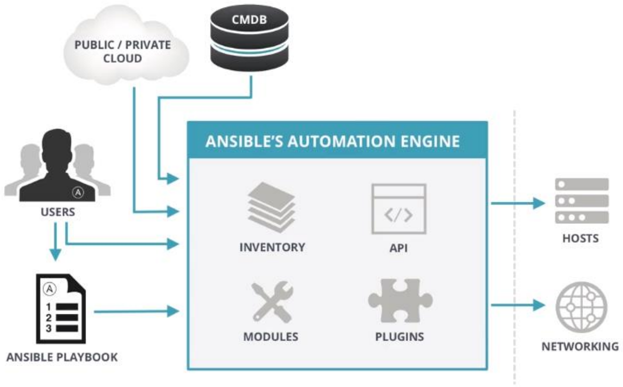


Figure 8: Ansible Engine Architecture

Red Hat Ansible Engine is powerful in that it can automate various IT processes such as application deployment, application configuration management, provisioning, continuous delivery, security and compliance, workflow orchestration, and orchestration of the application lifecycle. Ansible is an agentless architecture that uses OpenSSH and WinRM. Since there are no agents to exploit or update on the targeted systems, it is more efficient and more secure. The capability to execute a playbook against a system would be dependent on the access rights granted for execution. Ansible Engine, through orchestration, can be used to automate complex, multi-tiered deployments with ease.

Many businesses may have historically performed tasks manually such as installation and removal of services, installation and configuration of applications, updating of applications or packages, and so forth. If there are only a few servers on which to execute various tasks, it can be very simple for the administrator to log on to the server, execute the tasks, and log off. With larger deployments or as the IT operations grow, the time it can take to accomplish these tasks manually can be difficult. Moreover, the likelihood of missing manual processes for individual servers has the propensity to increase, especially as more administrators get involved in carrying out tasks simultaneously. Automation can help speed up the process and improve consistency for carrying out common tasks across large, growing, dynamic, and complex environments.

Red Hat Ansible Tower is an enterprise framework for controlling, securing, and managing an entity's automation with a UI and restful API. Red Hat Ansible Tower layers in control, knowledge, and delegation on top of Ansible's simple, powerful, and agentless automation engine. Red Hat Ansible Tower centralizes the running of Ansible playbooks to make it easier to integrate Ansible into other systems or workflows required for things like continuous integration and delivery (CI/CD) or DevOps processes. The power of Ansible allows for administrators to build workflows to provision machines, apply base system configuration, and deploy and configure applications and application components. It is possible for each step in a system provisioning operation to be designed and executed by different teams. For instance, a CI/CD workflow could include tasks to build an application, deploy the application to a test environment, run tests, and automatically promote the application to production based on the assessed test results. Different playbooks can be setup for varying responses depending on the output of success or failure of previous playbooks in the workflow.

Delegation through role-based access control with Red Hat Ansible Tower allows the organization to limit who can run tasks against which servers and when. All Ansible playbook output is captured and stored in Red Hat Ansible Tower so that all actions can be audited and mapped back to users who ran what job, what systems were updated, and the output of each system. Job failures can be further investigated by reviewing the output of the failed job.

A COMBINED SOLUTION FOR SUPPORTING SECURITY AND COMPLIANCE OBJECTIVES

Each of the solutions are powerful and can aid an organization with managing the complexity of modern IT operations. Each provides a nuance that has a useful role to play in securing, managing, monitoring, and automating IT operations. These tools are integrated in Red Hat Enterprise environments. Deploying, configuring, and maintaining a secure and compliant environment from the platform through the application can be accomplished using Red Hat Satellite, and Red Hat Ansible Tower and Red Hat Ansible Engine. Compliance policy can be enforced through control enablement capabilities of Red Hat Enterprise Linux Server and Red Hat Satellite using OpenSCAP to scan, detect, and remediate out of compliance deployments and configurations.

Red Hat Satellite can ensure that systems are patched and up to date to address vulnerabilities that could potentially be exploited by attackers. Red Hat Insights provides broader visibility into the state of an entity's systems and provides tailored proactive remediation steps to correct issues that may lead to inferior performance, instability, or security compromise. Ansible playbooks can be created to automate response to issues found in Red Hat Satellite and Red Hat Insights. Additionally, Red Hat Ansible Tower, using playbooks created in Red Hat Ansible Engine, can automate many routine tasks typically carried out by development and operations teams to enable greater consistency and improve efficiency. Red Hat Ansible Tower helps to scale IT automation by increasing control and security for these processes.

To help simplify and better secure Red Hat Enterprise infrastructures, Red Hat Enterprise Linux Server comes with Red Hat Enterprise Linux IdM. IdM provides the means to centrally manage users, machines, and groups in the Red Hat environment.

COALFIRE SCOPE AND APPROACH FOR REVIEW

Coalfire regularly produces opinion papers to guide a vendor's customers in understanding how the vendor's offering of products and services align with or provide support for security and compliance requirements. This paper is narrow in its scope in relation to a broader discussion of security and compliance. Therefore, it is not unlikely that this paper omits certain common security practices and procedures. Coalfire applies specific methodologies to the analysis of a vendor's products and services to determine the fitness of the same for inclusion in or applicability to various compliance programs.

SCOPE OF TECHNOLOGY AND STANDARD TO REVIEW

Coalfire was tasked by Red Hat to review Red Hat Satellite, Red Hat Insights, Red Hat Ansible Engine, Red Hat Ansible Tower, and Red Hat Enterprise Linux Server. The primary focus of this review was on current versions available at the time (April 2017) of the exercising of this engagement. Though it is anticipated that the findings of this paper will be relevant to future versions of these products and their general availability, the reader should consult with Red Hat to determine the applicability of specific product versions and to identify any changes as features and functions are added, modified, or removed. Included for review was the usefulness of these products to support native Red Hat infrastructures and platforms, including the use of Red Hat Enterprise Linux Server, available through subscription from supported cloud service providers. The review included the components, features, and functionality of the evaluated software.

Coalfire assumes that the product titles outlined in this paper are used in a manner where these technologies would be present in the scoped assessment environment, including both CDE systems and connected to, supporting, or security impacting systems. The technical evaluation assumed the use of Red Hat Enterprise Linux Server will be used for developing, managing, monitoring, orchestrating, automating, deploying, and hosting a payment entity's infrastructure and applications. It also assumes the adjacency of Red Hat Satellite, Red Hat Insights, Red Hat Ansible Engine, and Red Hat Ansible Tower to the CDE and used operationally in support of the management, visibility, and infrastructure and application lifecycle automation for systems in the CDE.

For this review, Coalfire included requirements from PCI DSS Requirements and Security Assessment Procedures version 3.2 April 2016 publication made available from https://www.pcisecuritystandards.org. For a broader understanding of the requirements and their applicability to technical solution implementation, Coalfire also reviewed supporting documentation provided by the PCI SSC including approach to assessment guidance, a sample report on compliance, and self-assessment questionnaires. Applied understanding of PCI DSS requirements and recommendations was also made available from PCI DSS guidance documents on relevant topics including, but not limited to, scoping and segmentation, best practices for maintaining PCI DSS requirements, risk assessment guidance, and PCI SSC guidance on cloud computing and virtualization.

Though the entirety of the PCI DSS v3.2 requirements were reviewed, the primary focus for review included technical controls that could be enabled or supported by the reviewed products and services. This does not diminish the importance of organizational or physical control requirements as part of a payment entity's responsibility to address.

COALFIRE EVALUATION METHODOLOGY

The PCI DSS requirements were reviewed and identified as either administrative or technical. A qualification of a requirement as administrative or technical was based on a combination of understanding of the requirement narrative, requirement testing procedures, and guidance. Additional insights supporting the understanding of requirements is further provided by additional guidance documents made available from PCI DSS, including, but not limited to, special interest group publications.

For this assessment, "non-technical", administrative requirements that include definition and documentation of policies, procedures, and standards were not considered applicable to the technical solution. Likewise, "non-technical" requirements including operational procedures that describe administrative processes that should not be automated were not assessed against the technology. Examples of this type of "non-technical" requirement include, but are not limited to, maintenance of facility visitor logs, verification of an individual's identity prior to granting physical or logical access, performance of periodic physical asset inventories, periodic log review, or generation of network topology and/or flow diagrams. Coalfire made exceptions for inclusion of administrative control requirements where the technology provided by Red Hat could inform or support an administrative procedure.

Technical requirements were then assessed to determine the applicability to the solution and/or solution components. Where achievement of the requirement objective was not likely to be met using the assessed products or services, the requirement was determined to be “not applicable” to the assessed technology. Examples of requirements that Coalfire determined not to be applicable to Red Hat Enterprise Linux, Red Hat Satellite, Red Hat Insights, Red Hat Ansible Engine, or Red Hat Ansible Tower included, among other things, use of encryption key management systems, wireless networking, technical physical access controls, or antivirus solutions. The reader will see, in some cases, a narrative was provided for some of these areas of control where the technology provided some minimal capability to support.

Where the requirement was determined to be applicable, Coalfire further assessed the capability and the extent of the solution to address the requirement. For applicable requirements, Coalfire designated a qualitative category of capability including whether the solution was determined to be able to fully support the requirement, partially support the requirement, or was unable to support the requirement. In the case where the technology was unable to support the requirement, the narrative may mention a third-party solution to be used in combination with the Red Hat technology to meet the control objectives. For example, the reviewed Red Hat technologies are all capable of producing event and audit logs to satisfy much of requirement 10; however, requirements to centrally store, protect, and retain logs is not a primary function of the Red Hat technologies. In this case, a third-party solution, such as log aggregation and/or SIEM, could be used to collect logs created by the Red Hat technologies to meet this control objective.

Coalfire’s methodology for review of the Red Hat products and services include review of product documentation including product data sheets, product specifications, release notes, installation guides, implementation and integration guides, administration guides, product use cases, and so forth provided by Red Hat and/or made available on Red Hat’s public web site. Interviews with Red Hat product subject matter experts help to fill in the gaps where product documentation does not provide specific insights to relevant capabilities or the product guide lacked detail to support fitness pertaining to compliance.

Additional Considerations

The findings of this opinion paper do not constitute certification of any product for use in a compliance program. This is outside the intent of PCI DSS to provide. However, this paper can be useful for helping a payment entity to consider the feasibility of using the reviewed products as part of their compliance program and to address needs and support organizational compliance initiatives.

The PCI DSS QSA may find useful information in this white paper and supporting materials to assist in their evaluation of the use of the included products and services during assessment activities that contribute to the production of a report on compliance (ROC). The guidance in this white paper and the supporting materials are intended to furnish a Coalfire opinion and are in no way meant to supplant or compromise the independent judgement required in performing PCI DSS assessments. In keeping with the PCI SSC Code of Professional Responsibility, which requires QSA companies and employees to “…adhere to high standards of ethical and professional conduct…”, Coalfire supports and upholds independent QSA judgements that might differ from this opinion.

RED HAT APPLICABILITY TO PCI DSS V3.2

Each applicable requirement is described in the table in the following section. This table includes the findings of applicability along with a short narrative describing the capability. The table uses Harvey Balls to describe the qualitative applicability of the technology to the requirement as determined by Coalfire (see: https://en.wikipedia.org/wiki/Harvey_Balls). This is not intended to be used as check boxes for organizations determining how to be compliant; rather, the following section is meant to provide guidance and understanding relative to the usefulness of the evaluated Red Hat products in a PCI DSS compliant environment.

The following key describes the meaning for the Harvey Balls.

DESIGNATION	KEY	MEANING
Fully Supported	●	The evaluated technology has the means, through configuration or by design, to fully support the requirement.
Partially Supported	◐	The evaluated technology has the means, through configuration or by design, to partially support the requirement. Additional technology or organizational policy, procedures, or standards may be required to fully meet the requirement objective.
Not Supported	○	The evaluated technology does not have the means, through configuration or by design, to support or meet the requirement objective. The technology relies on an external technology, system, or organizational procedure to meet the requirement objective.
Does Not Apply	N/A	The requirement, though it may be technical in nature, is not applicable to the evaluated technology. Meeting the requirement is completely achieved independently of the evaluated technology.
Non-Tech	Non-Tech	The requirement is purely operational requiring a defined and documented policy, procedure, or standard. This may also include operational, non-technical procedures such as the implementation of training, maintaining of paper logs, or performance of identity verification for personnel hiring practices.

Table 2 - Harvey Balls Key

For the sake of brevity, requirements that were found to not apply to the technology or were considered non-technical with no support provided by the technology were left off the following detail.

REQUIREMENT		NARRATIVE
1.2 Build firewall and router configurations that restrict connections between untrusted networks and any system components in the cardholder data environment. Note: An “untrusted network” is any network that is external to the networks belonging to the entity under review, and/or which is out of the entity's ability to control or manage.	◐	This requirement traditionally pertains to the network edge security mechanism for the payment entity. However, pertaining to internal network security and establishing secure boundaries between internal network zones or the segmenting of networks for scope reduction, firewalld or iptables host-based firewalls can be used to establish rules to establish zones of trust on the inside of a payment entity’s network to isolate workloads. Pertaining to the establishment of security zones for the isolation of workloads, Coalfire recommends using network segmentation techniques, either physical or virtual, to provide an additional network boundary between the CDE systems and non-CDE systems.

REQUIREMENT		NARRATIVE
1.2.1 Restrict inbound and outbound traffic to that which is necessary for the cardholder data environment, and specifically deny all other traffic.	◐	As an extension to the 1.2 narrative, firewalld or iptables are host-based firewalls that can be useful as a security mechanism to protect workloads on a host-by-host basis. firewalld can be enabled to deny all traffic, except where permitted based on trust zone defaults or policy exception in a defined ruleset. The design and implementation of firewall rulesets should be in alignment with entity-defined standards that provide a framework for determining what is necessary for proper function of the CDE systems. To fully address the requirement, it is recommended that the payment entity establish traditional edge boundary protections to secure the controlled network from the uncontrolled network or Internet.
1.2.2 Secure and synchronize router configuration files.	◐	firewalld allows for changes to the rulesets to be immediately available in runtime. Changes are simply added to the runtime ruleset. This functionality is useful for testing configuration changes as well as adding rules without having to restart services or drop connections. Once the firewall configuration change has been successfully tested, it is recommended that the change be permanently added to the ruleset in permanent mode. This ensures that the rule is a part of the ruleset after a reboot of the server or restart of the firewall daemon. To provide consistency for implementation of firewalld across all the payment entity's hosts, Red Hat Satellite can manage the hosts to provide consistent deployment of configuration and policy. Physical firewall and router configurations pertaining to the edge protection devices will require additional management and configuration by the payment entity to fully address this requirement.
1.3.1 Implement a DMZ to limit inbound traffic to only system components that provide authorized publicly accessible services, protocols, and ports.	◐	firewalld can be configured and enabled on hosts within the payment entity's network including hosts in a DMZ. The configuration of firewalld can help to establish trust boundaries for security zones within the network. A DMZ zone can be established with firewalld to support authorized communications for publicly accessible services, protocols, and ports. firewalld rulesets can be established by the payment entity to control the flow of information to and from, as well as between the DMZ hosts. To fully support the requirement, it is recommended to utilize traditional edge protection techniques to control traffic at the edge of the payment entity's network.

REQUIREMENT		NARRATIVE
1.3.4 Do not allow unauthorized outbound traffic from the cardholder data environment to the Internet.	◐	firewalld can be useful as in 1.3.1 to provide an additional layer of protection above that which is provided by traditional network security approaches. Rules can be enabled to prevent outbound traffic from hosts that have been assigned to a zone of trust to the Internet. Layering security in this way can be useful to limit impacts of misconfiguration or compromise elsewhere.
1.3.6 Place system components that store cardholder data (such as a database) in an internal network zone, segregated from the DMZ and other untrusted networks.	◐	Typically, separation of a system to a separate zone involves use of routing techniques with access control lists to create trust boundaries. This is often done using VLAN segmentation or the use of uniquely routed and controlled subnets to service each zone based on classification. firewalld can be used to apply additional layers of isolation for components of the CDE, that are more sensitive, such as databases, into separate internal network zones. Rulesets can be deployed to various servers in the CDE to restrict access to components of greater sensitivity to only that which is required.

REQUIREMENT		NARRATIVE
2.1 Always change vendor-supplied defaults and remove or disable unnecessary default accounts before installing a system on the network. This applies to ALL default passwords, including but not limited to those used by operating systems, software that provides security services, application and system accounts, point-of-sale (POS) terminals, payment applications, Simple Network Management Protocol (SNMP) community strings, etc.).	●	Red Hat Enterprise Linux Server hardening guide provides instructions for securely setting up the operating system. This includes guidance for setting initial root and/or administrator user accounts. The hardening guide also provides instructions for the secure set up of services that may be required for the function of the server or may be required for supporting systems. For checking that the configuration of the server is in alignment with a set of requirements, such as PCI DSS v3.2, Red Hat Enterprise Linux Server comes with OpenSCAP. The use of OpenSCAP can identify server configuration issues that could negatively impact the compliant state of the server. Rather than deploying and configuring servers manually on a server by server basis, large and complex environments can utilize Red Hat Satellite to centrally manage deployments and configuration. Red Hat Satellite can be connected to various content sources as specified by the entity. A required content source is the Red Hat Customer Portal, where Satellite may obtain digitally signed versions of Red Hat software, including ISOs, packages, updates, security patches, and so forth for which the entity has subscribed. Other content sources can include Puppet, Docker, GIT, and SCAP, as well as custom content sources integrated by the customer. Protected and approved system images or templates can be created and deployed from Red Hat Satellite with necessary hardening enabled at the time of deployment. Red Hat Satellite can be further used to perform any post deployment configuration or provisioning. System scanning and configuration checking, when combined with OpenSCAP, through Red Hat Satellite can identify configuration issues that may negatively impact compliance.

REQUIREMENT		NARRATIVE
2.2 Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards. Sources of industry-accepted system hardening standards may include, but are not limited to: • Center for Internet Security (CIS) • International Organization for Standardization (ISO) • Sysadmin Audit Network Security (SANS) Institute • National Institute of Standards Technology (NIST).	●	Red Hat provides well thought out system hardening guidelines for Red Hat Enterprise Linux Server. These hardening guidelines take into consideration many industry accepted hardening standards and guidelines. For configuration and verification, Red Hat Enterprise Linux Server comes with OpenSCAP. SCAP is a standardized checking solution for enterprise-level Linux infrastructure and is a line of specifications maintained by NIST for maintaining system security for enterprise systems. OpenSCAP is an auditing tool that contains security guidance content for various compliance frameworks. Content for OpenSCAP can be customized to meet the entity's specific standards, can make use of more common specifications, or can include content from the Red Hat user community. The SCAP workbench provides a GUI interface for scanning systems for analysis against pre-defined and customizable compliance content. OpenSCAP can be used for continuous scanning or scans can be performed on a scheduled basis. Rather than manage systems on a server-by-server basis, Red Hat Satellite enables the entity to centralize OpenSCAP scanning and reporting for registered servers. When combined with other tools such as Red Hat Insights and Red Hat Ansible Automation, configuration drift remediation can be automated to better support continuous compliance.
2.2.1 Implement only one primary function per server to prevent functions that require different security levels from co-existing on the same server. (For example, web servers, database servers, and DNS should be implemented on separate servers.) Note: Where virtualization technologies are in use, implement only one primary function per virtual system component.	●	Payment entities can utilize Red Hat Satellite to manage systems more strategically throughout their entire lifecycle. This includes the selection of content provided from Red Hat Satellite for deploying platforms and applications. Content can be made available to operations groups as views. The payment entity can improve control of system deployment and configuration, allowing for greater consistency in alignment with predefined standards and security design principles. Similarly, Red Hat Ansible Tower can support access controls and rules to ensure that deployments of applications into the environment stay consistent with standards and can only be performed by authorized personnel.

REQUIREMENT		NARRATIVE
2.2.2 Enable only necessary services, protocols, daemons, etc., as required for the function of the system.	●	Most network services installed with Red Hat Enterprise Linux Server are turned off by default. Services must be added to the server as needed. A part of configuration hardening, Red Hat Satellite can be used to deploy systems with only necessary services, protocols, daemons, etc., required for the function of the server. When combined with Red Hat Ansible Tower, Ansible playbooks can be automated to either isolate systems that have drifted from the standard or can be triggered to automatically correct the configuration drift.
2.2.3 Implement additional security features for any required services, protocols, or daemons that are considered to be insecure. Note: Where SSL/early TLS is used, the requirements in Appendix A2 must be completed.	●	Red Hat Enterprise Linux Server hardening guide identifies services, protocols, or daemons that are considered insecure. There are a variety of reasons a service or daemon may be considered insecure. These range from weak authentication mechanisms, authentication that is transmitted over the wire in clear text, or data that is sent over the wire in clear text. Though it is advised to avoid using insecure services, protocols, or daemons, there may be use cases where their enablement is difficult to avoid. In these cases, Red Hat Enterprise Linux Server includes security features that can be used for securing insecure services, protocols, or daemons. At a minimum, this includes encapsulating the communications using these services using TLS 1.2 encryption to prevent application data, authentication data, etc., from being transmitted in clear text. firewalld can be used to limit the source and destination targets for use of the insecure services, protocols, or daemons. Additional configuration of access controls, including the use of multifactor authentication, can be required to limit access to the insecure services. Moreover, SELinux can be deployed to limit the impact that compromised services can have on the server by confining processes to their own domain. Red Hat Enterprise Linux Server hardening guides also provide information for hardening and securing other services that may be required for the function of the server. This guidance is purposed to address risks to services such as Denial of Service Attacks (DoS), Distributed Denial of Service Attacks (DDoS), Script Vulnerability Attacks, and Buffer Overflow Attacks. Red Hat Insights can be used to identify where insecure services, protocols, and/or daemons are used without the necessary security required to protect them. Red Hat Insights categorizes impactful configuration issues by priority to allow security operations personnel to respond to issues of greatest concern first.

REQUIREMENT		NARRATIVE
2.2.4 Configure system security parameters to prevent misuse.	●	Red Hat Enterprise Linux Server comes with SELinux enabled by default. A benefit of SELinux is that processes and files are labeled with a type. A type defines a domain for a process and a type for files. Processes are separated from each other by running in their own domains, and SELinux policy rules define how processes can interact with files, as well as how processes can interact with each other. Access is only allowed if an SELinux policy rule exists that specifically allows it. In this way, a vulnerable or compromised process or service can be limited in its overall impact on the server to prevent misuse.
2.2.5 Remove all unnecessary functionality, such as scripts, drivers, features, subsystems, file systems, and unnecessary web servers.	●	Red Hat Enterprise Linux Server can be deployed with minimal functionality, including scripts, packages, drivers, features, subsystems, file systems, and unnecessary web services. The server can be configured with these as needed and as defined as part of the server's purpose and stated in the organization's standards guidance. Red Hat further provides further hardening guidance. Red Hat Satellite can be used to ensure that systems are in alignment with expected configurations and properly hardened. Remediation can be automated through Red Hat Satellite and/or Red Hat Ansible Tower to address configuration drift as necessary. Red Hat Insights can assist the organization with identification of any scripts, drivers, features, or other components that might pose a risk to the security of the managed system.
2.3 Encrypt all non-console administrative access using strong cryptography. Note: Where SSL/early TLS is used, the requirements in Appendix A2 must be completed.	●	Red Hat Enterprise Linux Server supports setting up non-console administration through SSH with OpenSSH to use strong cryptography including TLS 1.2 at a minimum. SSH with OpenSSH supports aes128-ctr/cbc and 3des-cbc, each are considered strong cryptography and viable options for non-console access to Linux servers. Red Hat Satellite can be used to ensure that Red Hat systems are enabled with proper access controls and configuration settings. Red Hat Ansible Automation can be used to remediate systems that are found to be out of compliance by automating the creation of Ansible playbooks to correct issues.
2.4 Maintain an inventory of system components that are in scope for PCI DSS.	●	Red Hat Satellite can be used to collect and maintain an inventory of all Red Hat systems and system components. Reports can be generated, including an inventory report, from Red Hat Satellite to inform the PCI DSS assessment process.

REQUIREMENT		NARRATIVE
3.4.1 If disk encryption is used (rather than file or column-level database encryption), logical access must be managed separately and independently of native operating system authentication and access control mechanisms (for example, by not using local user account databases or general network login credentials). Decryption keys must not be associated with user accounts. Note: This requirement applies in addition to all other PCI DSS encryption and key- management requirements.	◐	Red Hat Enterprise Linux Server provides disk encryption, including LUKS, allowing for encryption of partitions on the Linux computer. The default cipher used for LUKS is aes-cbc-essiv:sha256. The default key for LUKS is 256 bits. Ciphers that are available for LUKS is AES, Twofish, Serpent, cast5, and cast6. Red Hat Enterprise Linux Server 7.5 supports NBDE. Encryption and decryption processes are separate from authentication and access control mechanisms for Red Hat Enterprise Linux Server. The decryption keys for disk encryption can be separate from user accounts.
5.1 Deploy anti-virus software on all systems commonly affected by malicious software (particularly personal computers and servers).	○	Red Hat Enterprise Linux Server would require a third-party antivirus solution to be used to meet this control requirement. Anti-virus or anti-malware is not a feature or function of Red Hat Enterprise Linux Server.
6.1 Establish a process to identify security vulnerabilities, using reputable outside sources for security vulnerability information, and assign a risk ranking (for example, as “high,” “medium,” or “low”) to newly discovered security vulnerabilities. Note: Risk rankings should be based on industry best practices as well as consideration of potential impact. For example, criteria for ranking vulnerabilities may include consideration of the CVSS base score, and/or the classification by the vendor, and/or type of systems affected. Methods for evaluating vulnerabilities and assigning risk ratings will vary based on an organization’s environment and risk- assessment strategy. Risk rankings should, at a minimum, identify all vulnerabilities considered to be a “high risk” to the environment. In addition to the risk ranking, vulnerabilities may be considered “critical” if they pose an imminent threat to the environment, impact critical systems, and/or would result in a potential compromise if not addressed. Examples of critical systems may include security systems, public-facing devices and systems, databases, and other systems that store, process, or transmit cardholder data.	◐	Red Hat Insights, Red Hat Satellite, and Red Hat Enterprise Linux can enable OpenSCAP as a source to identify security vulnerabilities. OpenSCAP uses reputable sources to populate the databases of security vulnerability information. Red Hat Satellite can check the inventory of Red Hat Enterprise Linux system against known vulnerabilities as well as determine whether there are available security patches from Red Hat to remediate the vulnerabilities. Red Hat Insights can check the overall configuration of a system for vulnerability, configuration, and performance issues. Vulnerabilities, misconfigurations, and other security issues can be remediated through integration with Red Hat Ansible Automation. Playbooks are created to address detected and identified vulnerabilities that can be automatically run against targeted systems to correct and remediate issues.

REQUIREMENT		NARRATIVE
6.2 Ensure that all system components and software are protected from known vulnerabilities by installing applicable vendor-supplied security patches. Install critical security patches within one month of release. Note: Critical security patches should be identified according to the risk ranking process defined in Requirement 6.1.	●	Red Hat Satellite performs checks against the inventory of Red Hat systems against available Red Hat supplied security patches. Patching can be automated or scheduled based on the organization's requirements.
6.4.6 Upon completion of a significant change, all relevant PCI DSS requirements must be implemented on all new or changed systems and networks, and documentation updated as applicable. Note: This requirement is a best practice until January 31, 2018, after which it becomes a requirement.	◐	Changes to the Red Hat Enterprise Linux Server based environment can be centrally managed using Red Hat Satellite. Approved changes can be implemented through Red Hat Satellite such that all targeted Red Hat Satellite managed systems in inventory receive the change in a controlled manner. Compliance checks can be performed using OpenSCAP after changes are implemented to ensure that compliance policy is properly implemented and the change did not negatively impact the organization's compliance posture. The organization will be responsible for providing documentation for the change, including updating any topography and flow diagrams that may require updating because of the change. Some information for documentation may be acquired through reporting capabilities of Red Hat Insights and Red Hat Satellite, where a history of changes to the environment are captured.
7.2 Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed. This access control system(s) must include the following:	●	Red Hat Enterprise Linux Server includes IdM. Red Hat Enterprise Linux IdM provides a centralized and unified way to manage identity stores, authentication, policies, and authorization policies in a Linux-based domain. This allows for central management of access policy for Linux systems, rather than managing systems on a server-by-server basis. In this way, policies can be centrally managed and audited to ensure that access for devices and users is restricted based on need to know and denies access unless otherwise specifically allowed. The granularity of control for access can be increased when SELinux is used by enforcing mandatory access control where user or device activities can only operate in approved process domains.
7.2.1 Coverage of all system components.	●	Red Hat Enterprise Linux IdM allows for central management of access control for Red Hat Enterprise Linux Servers. Red Hat Enterprise Linux IdM has native integration capabilities with Windows Active Directory to support greater cohesiveness for access management across the enterprise where a mix of Windows and Linux hosts are deployed.

REQUIREMENT		NARRATIVE
7.2.2 Assignment of privileges to individuals based on job classification and function.	●	Greater control for assignment of privileges can be enabled for Red Hat Enterprise Linux Servers using Red Hat Enterprise Linux IdM. With Red Hat Enterprise Server IdM, administrations can maintain identities for Red Hat Enterprise Linux Servers in a central place, on the Red Hat Enterprise Linux IdM server. This allows policies to be applied uniformly to multiple machines at the same time. Access levels can be set for users by using host-based access control, delegation, or other rules. Privileged escalation, such as the use of sudo, can be managed centrally with Red Hat Enterprise Linux IdM. Additionally, the mounting of home directories for users can be defined on a user-by-user basis or based on role or group.
7.2.3 Default “deny-all” setting.	●	Improved and more cohesive central identity and access management can be achieved with Red Hat Enterprise Linux IdM. Red Hat Enterprise Linux IdM has a specific schema that defines a set of entries relevant to its purpose, such as entries for user or machine identities. Red Hat Enterprise Linux IdM is typically used as the identity and authentication server to manage identities within the boundaries of an enterprise or a project. The centralized management capabilities of Red Hat Enterprise Linux IdM and the native integration with Red Hat Enterprise Linux Server reduces the likelihood of unique or one-off server instances in the environment.
8.1.2 Control addition, deletion, and modification of user IDs, credentials, and other identifier objects.	●	Red Hat Enterprise Linux IdM can be used to control the addition, deletion, and modification of user IDs, credentials, and other identifier objects. Red Hat Enterprise Linux IdM can also be integrated with Windows Active Directory. User account credentials can also be centrally managed through Active Directory; whereas, access controls for authorization for these user accounts to Linux systems can be managed through Red Hat Enterprise Linux IdM. Red Hat Enterprise Linux IdM provides granular control for managing the lifecycle of user and device access control.
8.1.6 Limit repeated access attempts by locking out the user ID after not more than six attempts.	●	Policies can be established and applied from Red Hat Enterprise Linux IdM for Red Hat Enterprise Linux Servers to manage the authentication process such that invalid access attempts lock the user’s access and prevents any further access after not more than six failed attempts.
8.1.7 Set the lockout duration to a minimum of 30 minutes or until an administrator enables the user ID.	●	Policies can be established and applied from Red Hat Enterprise Linux IdM for Red Hat Enterprise Linux Servers to set a lockout duration for failed login attempts.

REQUIREMENT		NARRATIVE
8.1.8 If a session has been idle for more than 15 minutes, require the user to re-authenticate to re-activate the terminal or session.	●	Session duration policies can be established and applied with Red Hat Enterprise Linux IdM for users accessing Red Hat Enterprise Linux Servers with a threshold for idle session timeout. Local policies of Red Hat Enterprise Linux Servers can be centrally managed with Red Hat Satellite to ensure consistency throughout the enterprise.
8.2 In addition to assigning a unique ID, ensure proper user-authentication management for non-consumer users and administrators on all system components by employing at least one of the following methods to authenticate all users: • Something you know, such as a password or passphrase • Something you have, such as a token device or smart card • Something you are, such as a biometric.	●	Utilizing Red Hat Enterprise Linux IdM allows for greater granularity and central management of control for policies to be applied, including requirements for password complexity. Additionally, Red Hat Enterprise Linux IdM supports integration with and use of multi-factor authentication methods such as smart cards or token devices.
8.2.1 Using strong cryptography, render all authentication credentials (such as passwords/phrases) unreadable during transmission and storage on all system components.	●	Red Hat Enterprise Linux Server and Red Hat Enterprise Linux IdM supports the use of strong cryptography for transmission and storage of credentials.
8.2.3 Passwords/passphrases must meet the following: • Require a minimum length of at least seven characters. • Contain both numeric and alphabetic characters. Alternatively, the passwords/ passphrases must have complexity and strength at least equivalent to the parameters specified above.	●	Utilizing Red Hat Enterprise Linux IdM allows for greater granularity and central management of control for policies to be applied, including requirements for password complexity.
8.2.4 Change user passwords/passphrases at least once every 90 days.	●	Utilizing Red Hat Enterprise Linux IdM allows for greater granularity and central management of control for policies to be applied, including requirements for password complexity. This includes policies for defining the frequency with which passwords must be changed.
8.2.5 Do not allow an individual to submit a new password/passphrase that is the same as any of the last four passwords/passphrases he or she has used.	●	Utilizing Red Hat Enterprise Linux IdM allows for greater granularity and central management of control for policies to be applied, including requirements for password complexity. This includes limiting re-use of previously used passwords.

REQUIREMENT		NARRATIVE
8.3 Secure all individual non-console administrative access and all remote access to the CDE using multi-factor authentication. Note: Multi-factor authentication requires that a minimum of two of the three authentication methods (see Requirement 8.2 for descriptions of authentication methods) be used for authentication. Using one factor twice (for example, using two separate passwords) is not considered multi-factor authentication.	●	Red Hat Enterprise Linux IdM supports the use of and integration with multi-factor authentication methods including the additional use of a one-time password, using multiple methods, user-provided token, and/or smart card integration. The entity must include the additional factors for authentication for integration with Red Hat Enterprise Linux IdM capabilities.
8.3.1 Incorporate multi-factor authentication for all non-console access into the CDE for personnel with administrative access.	●	Red Hat Enterprise Linux IdM supports the use of and integration with multi-factor authentication methods including the additional use of a one-time password, using multiple methods, user-provided token, and/or smart card integration. The entity must include the additional factors for authentication for integration with Red Hat Enterprise Linux IdM capabilities.
8.3.2 Incorporate multi-factor authentication for all remote network access (both user and administrator, and including third-party access for support or maintenance) originating from outside the entity’s network.	◐	Red Hat Enterprise Linux IdM supports the use of and integration with multi-factor authentication methods including the additional use of a one-time password, using multiple methods, user-provided token, and/or smart card integration. The entity must include the additional factors for authentication for integration with Red Hat Enterprise Linux IdM capabilities.
8.5 Do not use group, shared, or generic IDs, passwords, or other authentication methods as follows: • Generic user IDs are disabled or removed. • Shared user IDs do not exist for system administration and other critical functions. • Shared and generic user IDs are not used to administer any system components.	◐	The use of Red Hat Enterprise Linux IdM as a central identity, authentication and authorization source for Red Hat Enterprise Linux Servers reduces the likelihood of the existence of unique or one-off hosts in the environment, where policy violations for the use of generic IDs or shared accounts can go undetected. It is up to the entity to follow this requirement.
10.1 Implement audit trails to link all access to system components to each individual user.	●	All the evaluated solutions provide mechanisms for the creation of audit records for all access to system components. The audit records are attributable to the individual user or subject acting upon the system components or objects.

REQUIREMENT		NARRATIVE
10.2.1 All individual user access to cardholder data.	◐	As it pertains to access to systems that contain CHD or that host payment applications, Red Hat Enterprise Linux Server is capable of being configured to log all access related to the server system itself. The audit logging capabilities should support audit and accountability requirements. Access to cardholder data is also often performed programmatically through application interfaces. In these cases, the other components involved with providing access to users for activities related to processing, storing, or transmission of CHD must also be recorded. These components can include the payment application itself, databases, storage subsystems, and so forth.
10.2.2 All actions taken by any individual with root or administrative privileges.	●	Red Hat Enterprise Linux Server is capable of being configured to log all actions taken by individuals with root or administrative privileges on the Red Hat Enterprise Linux Server. Red Hat Insights should be able to confirm that configuration settings are as planned for each system to verify that audit logging is enabled and to the desired level. Deviations from standards and policies could be addressed using Ansible. Ansible Automation can automate the creation and execution of playbooks to address deviations from the standard.
10.2.3 Access to all audit trails.	●	Red Hat Enterprise Linux Server is capable of being configured to log actions taken by individuals, including access to audit trails that are created and/or contained on a system. It is, however, recommended that logs generated by individual systems be collected centrally by a log aggregation server, syslog server, and/or SIEM for improved management capability.
10.2.4 Invalid logical access attempts.	●	As it pertains to accessing Red Hat Enterprise Linux Servers either locally or remotely, invalid login attempts can be configured to be logged.
10.2.5 Use of and changes to identification and authentication mechanisms—including but not limited to creation of new accounts and elevation of privileges—and all changes, additions, or deletions to accounts with root or administrative privileges.	●	Red Hat Enterprise Linux Server can be configured to log the use and changes to identification mechanisms. Red Hat Enterprise Linux IdM also logs the use and changes to identification and authentication mechanisms where IdM is being used as the central authority for identity and authentication for Red Hat Enterprise Server systems.
10.2.6 Initialization, stopping, or pausing of the audit logs.	●	Red Hat Enterprise Linux Server can be configured to log changes to the logging mechanisms within the system, including initializing, stopping, or pausing of audit logs. It is, however, recommended to use a centralized log collection and management tool.

REQUIREMENT		NARRATIVE
10.2.7 Creation and deletion of system-level objects.	●	Red Hat Enterprise Linux Server can be configured to log the creation and deletion of system-level objects. Red Hat Insights maintains a history of all changes that occur to a system that can be reviewed as well.
10.3.1 User identification.	●	Logs created by Red Hat systems include the identification of the user or process that took the action.
10.3.2 Type of event.	●	Logs created by Red Hat systems include the type of event that occurred.
10.3.3 Date and time.	●	Logs created by Red Hat systems include the data and time the event occurred.
10.3.4 Success or failure indication.	●	Logs created by Red Hat systems include whether the action taken was successful or failed.
10.3.5 Origination of event.	●	Logs created by Red Hat systems include the origination of the event where it occurred.
10.3.6 Identity or name of affected data, system component, or resource.	●	Logs created by Red Hat systems include the identity or name of affected data, system component, and/or resource.
10.4 Using time-synchronization technology, synchronize all critical system clocks and times and ensure that the following is implemented for acquiring, distributing, and storing time. Note: One example of time synchronization technology is Network Time Protocol (NTP).	●	All Red Hat systems can be configured to synchronize with a common time source such as NTP pool.
10.4.1 Critical systems have the correct and consistent time.	●	All Red Hat systems can be configured to synchronize with a common time source such as an NTP pool and can be configured to maintain a level of synchronization within desired thresholds to establish a consistent and correct time.
10.4.2 Time data is protected.	●	All Red Hat systems can be configured to have time data and time data configuration protected using role-based access controls to limit access to the time data and timekeeping mechanisms to authorized personnel. Changes to time keeping can be logged with notifications when unexpected events occur.
10.4.3 Time settings are received from industry-accepted time sources.	●	Red Hat systems can be configured to use time settings from industry-accepted time sources. The time sources can be configured by the customer. This can also be centrally managed and configured for each managed system using Red Hat Satellite.

REQUIREMENT		NARRATIVE
10.5 Secure audit trails so they cannot be altered.	◐	All Red Hat systems create audit logs. The logs can be configured to be collected by or sent to a SIEM solution or syslog server. To fully support the requirement, it is recommended to use a third-party tool to provide additional levels of security pertaining to audit trails, including protection of logs from alteration.
10.5.1 Limit viewing of audit trails to those with a job-related need.	◐	All Red Hat systems can be configured to limit access to view audit trails on a need to know basis, using role-based access controls. To fully support the requirement, it is recommended, however, to use a third-party solution, such as a SIEM, for the aggregation, security, and analysis of audit trails.
10.5.2 Protect audit trail files from unauthorized modifications.	◐	All Red Hat systems can be configured to limit access to view audit trails on a need to know basis, using role-based access controls and to prevent modification or alteration of generated audit records. Additional protections are built into Red Hat Enterprise Linux Server, such as mandatory access controls enforced with SELinux. To fully support this requirement, it is recommended, however, to use a third-party purpose-built solution, such as a SIEM, for the aggregation, security, and analysis of audit trails.
10.5.3 Promptly back up audit trail files to a centralized log server or media that is difficult to alter.	◐	All Red Hat systems can be configured to have the audit trail files backed up to a centralized log server or media that is difficult to alter. To fully support this requirement, it is recommended, however, to implement a third-party purpose-built solution, such as SIEM, as a central authority for logs, event, and audit trails for correlation and analysis.
10.5.4 Write logs for external-facing technologies onto a secure, centralized, internal log server or media device.	◐	All Red Hat systems can be configured to have the audit trail files backed up to an internal, centralized log server to reduce the risk that logs are lost or altered. To fully support this requirement, it is recommended, however, to implements a third-party purpose-built solution, such as SIEM, as a central authority for logs, event, and audit trails for correlation and analysis.

REQUIREMENT		NARRATIVE
10.5.5 Use file-integrity monitoring or change-detection software on logs to ensure that existing log data cannot be changed without generating alerts (although new data being added should not cause an alert).	○	The evaluated Red Hat products all have the capability of producing auditable events. These events can be gathered by third party log collection and aggregation solutions, or SIEM solutions. The integrity of collected logs is typically handled by third party purpose built solutions. It is recommended to use a third-party purpose-built solution, such as SIEM, to be a central authority for system generated logs, events, and audit trails. These systems take into consideration many compliance framework requirements regarding the confidentiality, integrity, and availability of audit data.

Table 3 - Requirements Matrix

COALFIRE CONCLUSION

No one product is capable of fully addressing security and compliance requirements. Security is a design principle that must be addressed through carefully planned and implemented strategies. Entities seeking compliance are best able to obtain it through a governance, risk, and compliance (GRC) program. For this reason, the introduction of new technologies in a compliant organization should include the payment entity-defined security design principles to reduce risk and maintain or improve security. The benefit in this case is the ability to apply the security design principles to increase day-to-day assurance of compliance, reduce risk, and improve security. While Coalfire disclaims the generic suitability of any product for regulatory compliance, Coalfire can confirm that, through careful planning, design, and implementation, Red Hat Enterprise Linux Server can be included as part of a payment entity's infrastructure in support of a PCI DSS compliant environment. Moreover, Red Hat Satellite, Red Hat Insights, and Red Hat Ansible Engine and Red Hat Ansible Tower can be useful tools to support compliance.

REFERENCES

These are referenced materials that Coalfire used to study the product(s) or solution(s) as well as the PCI DSS framework. These can be helpful for the reader to identify the basis for Coalfire's findings. These references are also useful for the reader to do their own due diligence or read beyond the material presented in this paper.

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/pdf/security_guide/Red_Hat_Enterprise_Linux-7-Security_Guide-en-US.pdf

https://www.redhat.com/cms/managed-files/li-rhel-7-identity-management-technology-brief-12069297-0514mm-en.pdf

https://access.redhat.com/documentation/en-us/red_hat_network_satellite/5.5/html/user_guide/chap-red_hat_network_satellite-user_guide-openscap

https://scap.nist.gov/specifications/ocil/

https://oval.mitre.org/

https://www.first.org/cvss/

https://nvd.nist.gov/vuln-metrics/cvss

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/security_guide/sec-using_firewalls

http://www.firewalld.org/

https://www.redhat.com/en/engage/red-hat-insights-infrastructure-20170721?sc_cid=701f2000000tgDCAAY

https://www.youtube.com/watch?v=XwD2eEreCoQ

www.pcisecuritystandards.org

PCI DSS v3.2 April 2016

BIBLIOGRAPHY

PCI Security Standards Council. (2016, December). *Information Supplement: Guidance for PCI DSS Scoping and Newtork Segmentation.* Retrieved from www.pcisecuritystandards.org: https://www.pcisecuritystandards.org

PCI Security Standards Council. (2016, April). *Payment Card Industry (PCI) Data Security Standard.* Retrieved from www.pcisecuritystandards.org: https://www.pcisecuritystandards.org

ABOUT THE AUTHORS

Jason Macallister | Senior Consultant, Cyber Engineering, Coalfire

Mr. Macallister contributes as an author and thought leader on information security and regulatory compliance topics for Coalfire's clientele relative to advanced infrastructures and emerging products and solutions.

Chris Krueger | Principal, Cyber Engineering, Coalfire

As Principle, Mr. Krueger contributes as an author and thought leader on information security and regulatory compliance topics for Coalfire's clientele in the “new and emerging” technology areas.

Fred King | Senior Consultant, Cyber Engineering, Coalfire

As a senior consultant, Mr. King is responsible for security architecture with a focus on applying defense in depth strategies to modern data center infrastructures. He is a leader, an experienced technologist, and a QSA in the payment card industry.

May 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

Coalfire is solely responsible for the contents of this document as of the date of publication. The contents of this document are subject to change at any time based on revisions to the applicable regulations and standards (HIPAA, PCI-DSS et.al). Consequently, any forward-looking statements are not predictions and are subject to change without notice. While Coalfire has endeavored to ensure that the information contained in this document has been obtained from reliable sources, there may be regulatory, compliance, or other reasons that prevent us from doing so. Consequently, Coalfire is not responsible for any errors or omissions, or for the results obtained from the use of this information. Coalfire reserves the right to revise any or all of this document to reflect an accurate representation of the content relative to the current technology landscape. In order to maintain contextual accuracy of this document, all references to this document must explicitly reference the entirety of the document inclusive of the title and publication date; neither party will publish a press release referring to the other party or excerpting highlights from the document without prior written approval of the other party. If you have questions with regard to any legal or compliance matters referenced herein you should consult legal counsel, your security advisor and/or your relevant standard authority.

Red Hat Product Applicability Guide for PCI DSS 3.2 May 2018