NIST SPECIAL PUBLICATION 1800-31

Improving Enterprise Patching for General IT Systems:
Utilizing Existing Tools and Performing Processes in Better Ways

Includes Executive Summary (A); Security Risks and Capabilities (B); and How-To Guides (C)

Tyler Diamond*
Alper Kerman
Murugiah Souppaya
Kevin Stine
Brian Johnson
Chris Peloquin
Vanessa Ruffin
Mark Simos
Sean Sweeney
Karen Scarfone

**Former employee; all work for this publication was done while at employer*

November 2021

DRAFT

This publication is available free of charge from
https://www.nccoe.nist.gov/projects/building-blocks/patching-enterprise

NIST SPECIAL PUBLICATION 1800-31

Improving Enterprise Patching for General IT Systems: Utilizing Existing Tools and Performing Processes in Better Ways

Includes Executive Summary (A); Security Risks and Capabilities (B); and How-To Guides (C)

Tyler Diamond*
Alper Kerman
Murugiah Souppaya
Kevin Stine
National Cybersecurity Center of Excellence
Information Technology Laboratory

Brian Johnson
Chris Peloquin
Vanessa Ruffin
The MITRE Corporation
McLean, VA

Mark Simos
Sean Sweeney
Microsoft
Redmond, WA

Karen Scarfone
Scarfone Cybersecurity
Clifton, Virginia

**Former employee; all work for this publication was done while at employer*

DRAFT

November 2021

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the non-exclusive functions and duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

NIST SPECIAL PUBLICATION 1800-31A

Improving Enterprise Patching for General IT Systems:

Utilizing Existing Tools and Performing Processes in Better Ways

Volume A:
Executive Summary

Alper Kerman
Murugiah Souppaya
Kevin Stine
National Cybersecurity Center of Excellence
Information Technology Laboratory

Mark Simos
Sean Sweeney
Microsoft
Redmond, Washington

Karen Scarfone
Scarfone Cybersecurity
Clifton, Virginia

DRAFT

This publication is available free of charge from
https://www.nccoe.nist.gov/projects/building-blocks/patching-enterprise

Executive Summary

For decades, cybersecurity attacks have highlighted the dangers of having computers with unpatched software. Even with widespread awareness of these dangers, however, keeping software up-to-date with patches remains a problem. Deciding how, when, and what to patch can be difficult for any organization. Each organization must balance security with mission impact and business objectives by using a risk-based methodology. To address these challenges, the NCCoE is collaborating with cybersecurity technology providers to explore approaches for improving enterprise patching practices for general information technology (IT) systems. These practices are intended to help your organization improve its security and reduce the likelihood of data breaches with sensitive personal information and other successful compromises. The practices can also play an important role as your organization embarks on a journey to zero trust.

CHALLENGE

There are a few root causes for many data breaches, malware infections, ransomware attacks, and other security incidents, and known—but unpatched—vulnerabilities in software is one of them. Implementing a few security hygiene practices, such as patching operating systems, applications, and firmware, can prevent many incidents from occurring, lowers the potential impact of incidents that do occur, and increases the cost to the attacker. Unfortunately, security hygiene is easier said than done. Despite widespread recognition that patching is effective and attackers regularly exploit unpatched software, many organizations do not adequately patch. There are myriad reasons why, not the least of which are that it's resource-intensive and that the act of patching can reduce system and service availability. Many organizations struggle to prioritize patches, test patches before deployment, and adhere to policies for how quickly patches are applied in different situations. Delaying patch deployment gives attackers a larger window of opportunity.

This practice guide can help your organization:

- overcome common obstacles involving enterprise patching for general IT systems
- achieve a comprehensive security hygiene program based on existing standards, guidance, and publications
- enhance its recovery from incidents that occur, and minimize the impact of incidents on the organization and its constituents

SOLUTION

To address these challenges, the NCCoE is collaborating with cybersecurity technology providers to develop an example solution. It will demonstrate how tools can be used to 1) implement the inventory and patching capabilities organizations need to handle both routine and emergency patching situations, as well as 2) implement workarounds, isolation methods, or other alternatives to patching. The solution will also demonstrate recommended security practices for patch management systems themselves.

The NCCoE is assembling existing commercial and open source tools to aid with the most challenging aspects of patching. The NCCoE is building upon previous NIST work documented in *NIST Special Publication (SP) 800-40 Revision 3, Guide to Enterprise Patch Management Technologies* and *NIST SP 800-184, Guide for Cybersecurity Event Recovery*.

Collaborator	Security Capability or Component
	Asset discovery and inventory; network access control; network policy enforcement
	Hardware and firmware inventory; firmware vulnerability assessment; firmware integrity monitoring; firmware software updates
	Asset discovery and inventory; security policy enforcement
	Asset inventory; configuration management; software updates; vulnerability scanning for source code as part of a DevOps pipeline
	Security policy enforcement; vulnerability scanning and reporting; software discovery and inventory; firmware vulnerability assessment and policy enforcement
	Asset discovery; configuration management; software updates
	Asset discovery and inventory; vulnerability scanning and reporting
	Vulnerability scanning and remediation; configuration management; software updates

While the NCCoE is using commercial and open source products to address this challenge, the practice guide will not endorse these particular products, nor will it guarantee compliance with any regulatory initiatives. Your organization's information security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of a solution.

HOW TO USE THIS GUIDE

Depending on your role in your organization, you might use this guide in different ways:

Business decision makers, including chief information security and technology officers can use this part of the guide, *NIST SP 1800-31A: Executive Summary*, to understand the drivers for the guide, the cybersecurity challenge we address, our approach to solving this challenge, and how the solution could benefit your organization. Business decision makers can also use *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology*. It complements the implementation focus of this guide by recommending creation of an enterprise strategy to simplify and operationalize patching while also reducing risk.

Technology, security, and privacy program managers who are concerned with how to identify, understand, assess, and mitigate risk can use *NIST SP 1800-31B: Security Risks and Capabilities,* which describes what we built and why, including the risk analysis performed and the security capabilities provided by the example implementation. *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology* may also be helpful.

IT professionals who want to implement an approach like this can make use of *NIST SP 1800-31C: How-To Guides*, which provide specific product installation, configuration, and integration instructions for building the example implementation, allowing you to replicate all or parts of this project.

SHARE YOUR FEEDBACK

You can view or download the guide at https://www.nccoe.nist.gov/projects/building-blocks/patching-enterprise. Help the NCCoE make this guide better by sharing your thoughts with us as you read the guide. If you adopt this solution for your own organization, please share your experience and advice with us. We recognize that technical solutions alone will not fully enable the benefits of our solution, so we encourage organizations to share lessons learned and best practices for transforming the processes associated with implementing this guide.

To provide comments or to learn more by arranging a demonstration of this example implementation, contact the NCCoE at cyberhygiene@nist.gov.

COLLABORATORS

Collaborators participating in this project submitted their capabilities in response to an open call in the Federal Register for all sources of relevant security capabilities from academia and industry (vendors and integrators). Those respondents with relevant capabilities or product components signed a Cooperative Research and Development Agreement (CRADA) to collaborate with NIST in a consortium to build this example solution.

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

NIST SPECIAL PUBLICATION 1800-31B

Improving Enterprise Patching for General IT Systems:

Utilizing Existing Tools and Performing Processes in Better Ways

Volume B:
Security Risks and Capabilities

Tyler Diamond*
Alper Kerman
Murugiah Souppaya
National Cybersecurity Center of Excellence
Information Technology Laboratory

Brian Johnson
Chris Peloquin
Vanessa Ruffin
The MITRE Corporation
McLean, Virginia

Karen Scarfone
Scarfone Cybersecurity
Clifton, Virginia

**Former employee; all work for this publication was done while at employer*

November 2021

DRAFT

This publication is available free of charge from
https://www.nccoe.nist.gov/projects/building-blocks/patching-enterprise

DISCLAIMER

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

National Institute of Standards and Technology Special Publication 1800-31B, Natl. Inst. Stand. Technol. Spec. Publ. 1800-31B, 49 pages, (November 2021), CODEN: NSPUE2

FEEDBACK

You can improve this guide by contributing feedback. As you review and adopt this solution for your own organization, we ask you and your colleagues to share your experience and advice with us.

Comments on this publication may be submitted to: cyberhygiene@nist.gov.

Public comment period: November 17, 2021 through January 10, 2022

All comments are subject to release under the Freedom of Information Act.

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Despite widespread recognition that patching is effective and attackers regularly exploit unpatched software, many organizations do not adequately patch. There are myriad reasons why, not the least of which are that it's resource-intensive and that the act of patching can reduce system and service availability. Also, many organizations struggle to prioritize patches, test patches before deployment, and adhere to policies for how quickly patches are applied in different situations. To address these challenges, the NCCoE is collaborating with cybersecurity technology providers to develop an example solution that addresses these challenges. This NIST Cybersecurity Practice Guide explains how tools can be used to implement the patching and inventory capabilities organizations need to handle both routine

and emergency patching situations, as well as implement workarounds, isolation methods, or other alternatives to patching. It also explains recommended security practices for patch management systems themselves.

KEYWORDS

cyber hygiene; enterprise patch management; firmware; patch; patch management; software; update; upgrade; vulnerability management

ACKNOWLEDGMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Peter Romness	Cisco
Matthew Hyatt	Cisco
John Loucaides	Eclypsium
Travis Raines	Eclypsium
Timothy Jones	Forescout
Tom May	Forescout
Michael Correa	Forescout
Jeffrey Ward	IBM MaaS360 with Watson
Joseph Linehan	IBM MaaS360 with Watson
Cesare Coscia	IBM MaaS360 with Watson
Jim Doran	IBM Research Team
Shripad Nadgowda	IBM Research Team
Victoria Mosby	Lookout

Name	Organization
Tim LeMaster	Lookout
Dan Menicucci	Microsoft
Steve Rachui	Microsoft
Parisa Grayeli	The MITRE Corporation
Yemi Fashina	The MITRE Corporation
Nedu Irrechukwu	The MITRE Corporation
Joshua Klosterman	The MITRE Corporation
Allen Tan	The MITRE Corporation
Josh Moll	Tenable
Chris Jensen	Tenable
Jeremiah Stallcup	Tenable
John Carty	VMware
Kevin Hansen	VMware
Rob Robertson	VMware
Rob Hilberding	VMware
Brian Williams	VMware

The Technology Partners/Collaborators who participated in this build submitted their capabilities in response to a notice in the Federal Register. Respondents with relevant capabilities or product components were invited to sign a Cooperative Research and Development Agreement (CRADA) with NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Build Involvement
Cisco	Cisco Firepower Threat Defense (FTD) Cisco Identity Services Engine (ISE)
Eclypsium	Eclypsium Administration and Analytics Service
Forescout	Forescout Platform
IBM	IBM Code Risk Analyzer IBM MaaS360 with Watson
Lookout	Lookout Mobile Endpoint Security
Microsoft	Microsoft Endpoint Configuration Manager
Tenable	Nessus Tenable.io Tenable.sc
VMware	VMware vRealize Automation SaltStack Config

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly to conform to the publication and from which no deviation is permitted. The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited. The terms “may” and “need not” indicate a course of action permissible within the limits of the publication. The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory (ITL) draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication

or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

1. under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or
2. without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to: cyberhygiene@nist.gov

Contents

List of Tables

1 Summary

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) recognizes the challenges that organizations face in keeping software up to date with patches. Patches correct security and functionality problems in software and firmware. From a security perspective, patches are most often of interest because they are mitigating software flaw vulnerabilities; applying patches to eliminate these vulnerabilities significantly reduces the opportunities for exploitation.

Patches serve other purposes than just fixing software flaws; they can also add new features to software and firmware, including security capabilities. Sometimes there are alternatives to patches, such as temporary workarounds involving software or security control reconfiguration, but these workarounds are not permanent fixes and they may impact functionality.

The NCCoE developed the Critical Cybersecurity Hygiene: Patching the Enterprise (Patching) project to provide approaches for improving enterprise patching practices for general information technology (IT) systems. The aim is to help organizations balance security with mission impact and business objectives.

This project utilizes commercial tools to aid with functions that include asset discovery characterization and prioritization, and patch implementation tracking and verification. It includes actionable and prescriptive guidance on establishing policies and processes for the entire patching lifecycle. This volume explains why we built the example solution to address patching challenges, including the risk analysis we performed and the security capabilities that the example solution provides.

1.1 Challenge

There are a few root causes for many data breaches, malware infections such as ransomware, and other security incidents, and known—but unpatched—vulnerabilities in software are one of them. Implementing a few security hygiene practices, such as patching operating systems, applications, and firmware, can address those root causes. That prevents many incidents from occurring by minimizing the attack surface and lowers the potential impact of incidents that occur. In other words, security hygiene practices make it harder for attackers to succeed and reduce the damage they can cause.

Unfortunately, security hygiene is easier said than done. Despite widespread recognition that (a) patching is effective and (b) attackers regularly exploit unpatched software, many organizations do not adequately patch. There are myriad reasons why, not the least of which are that it is resource-intensive and that the act of patching is perceived to reduce system and service availability. However, delaying patch deployment gives attackers a larger window of opportunity to take advantage of the exposure. Many organizations struggle to inventory their assets, prioritize patches, have defined and consistent process and procedures for deployment, and adhere to policies and metrics for how quickly patches are applied in different situations. Also, deploying enterprise patch management tools that operate with

privileged access within an enterprise can itself create additional security risks for an organization if the tools are not secured properly.

1.2 Solution

To address these challenges, the NCCoE is collaborating with cybersecurity technology providers to develop an example solution. It will demonstrate how tools can be used to 1) implement the inventory and patching capabilities organizations need to handle both routine and emergency patching situations, as well as 2) implement workarounds, isolation methods, or other alternatives to patching. The solution will also demonstrate recommended security practices for protecting the patch management systems themselves against threats.

This draft covers both phases of the example solution, which involves patching, updating, and configuring two types of general IT assets. Phase 1 focuses on desktop and laptop computers and on-premises servers, and phase 2 adds mobile devices and containers.

The NCCoE has also created a companion publication, *NIST Special Publication (SP) 800-40 Revision 4 (Draft),* *Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology*. It complements the implementation focus of this guide by recommending creation of an enterprise strategy to simplify and operationalize patching while also reducing risk.

1.3 Benefits

The demonstrated approach offers several benefits to organizations that implement it, including the following:

- Vulnerabilities in the organization's IT systems that are susceptible to cyber attacks are addressed more quickly, which reduces risk and lowers the likelihood of an incident occurring.
- Increased automation provides a traceable and repeatable process and leads to a decrease in hours worked by the organization's security administrators, system administrations, and others who have patching responsibilities.
- It improves compliance with laws, regulations, mandates, local organization policy, and other requirements to keep the organization's software patched.
- The practices it demonstrates can play an important role as your organization embarks on a journey to zero trust.

2 How to Use This Guide

This NIST Cybersecurity Practice Guide demonstrates a standards-based example solution and provides users with the information they need to replicate the proposed approach for improving enterprise

patching practices for general IT systems. This design is modular and can be deployed in whole or in part.

This guide contains three volumes:

- NIST SP 1800-31A: *Executive Summary* – why we wrote this guide, the challenge we address, why it could be important to your organization, and our approach to solving the challenge
- NIST SP 1800-31B: *Security Risks and Capabilities* – why we built the example implementation, including the risk analysis performed and the security capabilities provided by the implementation **(you are here)**
- NIST SP 1800-31C: *How-To Guides* – what we built, with instructions for building the example implementation, including all the details that would allow you to replicate all or parts of this project

Depending on your role in your organization, you might use this guide in different ways:

Business decision makers, including chief security and technology officers, will be interested in the *Executive Summary, NIST SP 1800-31A*, which describes the following topics:

- challenges that enterprises face in mitigating risk from software vulnerabilities
- example solution built at the NCCoE
- benefits of adopting the example solution

Business decision makers can also use *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology*.

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in this part of the guide, *NIST SP 1800-31B*, which describes what we did and why. The following sections will be of particular interest:

- Section 3.5.1, Threats, Vulnerabilities, and Risks, provides a description of the risk analysis we performed.
- Section 3.5.2, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices.

You might share the *Executive Summary, NIST SP 1800-31A*, with your leadership team members to help them understand the importance of adopting standards-based, automated patch management. Also, *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology* may be helpful to you and your leadership team.

IT professionals who may be interested in implementing an approach similar to ours will find the entire practice guide useful. In particular, the How-To portion of the guide, *NIST SP 1800-31C* could be used to replicate all or parts of the build created in our lab. Furthermore, the How-To portion of the guide

provides specific product installation, configuration, and integration instructions for implementing the example solution. We have omitted the general installation and configuration steps outlined in manufacturers' product documentation since they are typically made available by manufacturers. Instead, we focused on describing how we incorporated the products together in our environment to create the example solution.

This guide assumes that the reader of this document is a seasoned IT professional with experience in implementing security solutions within an enterprise setting. While we have used a suite of commercial products to address this challenge, this guide does not endorse these particular products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of an automated enterprise patch management system. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and recommended practices. Section 4.2, Technologies, lists the products we used and maps them to the cybersecurity controls provided by this example solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution, but a possible solution. This is a draft guide. We seek feedback on its contents and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to cyberhygiene@nist.gov.

2.1 Typographic Conventions

The following table presents typographic conventions used in this volume.

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For language use and style guidance, see the *NCCoE Style Guide*.
Bold	names of menus, options, command buttons, and fields	Choose **File** > **Edit**.
`Monospace`	command-line input, onscreen computer output, sample code examples, and status codes	`mkdir`
`Monospace Bold`	command-line user input contrasted with computer output	**`service sshd start`**

Typeface/Symbol	Meaning	Example
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

3 Approach

The NCCoE issued an open invitation to technology providers to participate in demonstrating how organizations can use technologies to improve enterprise patch management for their general IT assets. Cooperative Research and Development Agreements (CRADAs) were established with qualified respondents, and a build team was assembled. The team fleshed out the initial architecture, and the collaborators' components were composed into an example implementation, i.e., build. The build team documented the architecture and design of the build. As the build progressed, the team documented the steps taken to install and configure each component of the build.

Finally, the team verified that the build provided the desired capabilities. This included conducting a risk assessment and a security characteristic analysis, then documenting the results, including mapping the security contributions of the demonstrated approach to the *Framework for improving Critical Infrastructure Cybersecurity* (NIST Cybersecurity Framework), NIST SP 800-53, the *Security Measures for "EO-Critical Software" Use Under Executive Order (EO) 14028*, and other relevant standards and guidelines.

3.1 Audience

This guide is intended for chief information officers (CIOs), chief information security officers (CISOs), cybersecurity directors and managers, and others who are responsible for managing organizational risk related to patch management. It also contains information of use for security engineers and architects, system administrators, security operations personnel, and others who are involved in enterprise patch management.

3.2 Scope

This project only covers general IT systems: desktops/laptops, servers, virtual machines and containers, and mobile devices running current software. There are additional challenges with patching legacy IT systems, as well as industrial control systems (ICS), Internet of Things (IoT) devices, and other technologies stemming from operational technology (OT), so they will not be covered in this project.

All aspects of security hygiene other than those related to patching are out of the scope of this project.

3.3 Assumptions

This project is guided by the following assumptions:

- An IT endpoint for an enterprise would have firmware, operating system(s), and application(s) to be patched. The endpoint may be in a fixed location within the organization's own facilities or in a fixed location at a third-party facility (e.g., a data center), or it may be intended for use in multiple locations, such as a laptop used at the office and for telework. The proposed approach for improving enterprise patching practices would have to account for all these possibilities.
- Problems sometimes occur with patches, such as a failure during installation, a patch that cannot take effect until the endpoint is rebooted, or a patch that is uninstalled because of operational concerns or because an attacker rolled it back in order to have an entry point to the system. This project follows a "verify everything and trust nothing" philosophy that does not assume installing a patch automatically means the patch is successfully and permanently applied.
- There are no standard protocols, formats, etc. for patch management, including patch distribution, integrity verification, installation, and installation verification. It is also highly unlikely for a single patch management system to be able to handle all patch management responsibilities for all software on IT endpoints. For example, some applications may handle patching themselves and not be capable of integrating with a patch management system for patch acquisition and installation.

3.4 Scenarios

This project will address all the scenarios described below.

3.4.1 Scenario 0: Asset identification and assessment

This scenario identifies the assets and classifies them based on vulnerability impact levels to prioritize the order of remediation. It leverages tools to discover assets across the enterprise and the cloud and to enumerate their firmware, operating systems (OSes), and applications. Knowing which software and software versions are in use and predetermining remediation priorities are critically important to all other patching processes. Without accurate, up-to-date, and comprehensive information, an organization will have difficulties effectively and efficiently performing patching processes, thus increasing risk. While many enterprises have constant asset attrition, it is important to have full and accurate inventory of critical assets and the best possible inventory for the full enterprise.

3.4.2 Scenario 1: Routine patching

This is the standard procedure for patches that are on a regular release cycle and haven't been elevated to an active emergency status (see Scenario 3). Routine patching includes endpoint firmware, OS, and applications, and server OS and applications hosted on-premises or in the cloud (e.g., Infrastructure as a

Service). Most patching falls under this scenario or Scenario 2. However, because routine patching does not have the urgency of emergency patching, and routine patch installation can interrupt operations (e.g., device reboots), it is often postponed and otherwise neglected. This provides many additional windows of opportunity for attackers.

3.4.3 Scenario 2: Routine patching with cloud delivery model

This is the standard procedure for patches that are delivered through a cloud delivery model, such as a mobile device or a "Windows as a Service (WaaS)" model with Windows operating systems, Apple Software Update, and mobile device software updates for Android and iOS devices provided by device manufacturers or mobile operators. This scenario is similar in importance to Scenario 1, Routine Patching. However, organizations may not be as accustomed to cloud-delivered patches (which are frequently cumulative for the whole system vs. discrete patches), so this scenario is somewhat more likely to be overlooked by organizations, which increases risk.

3.4.4 Scenario 3: Emergency patching

This is the emergency procedure to address active patching emergencies in a crisis situation, such as extreme severity vulnerabilities like MS17-010, as well as vulnerabilities that are being actively exploited in the wild. The scope of targets is the same as scenario 1. Emergency patching needs to be handled as efficiently as possible to prevent imminent exploitation of vulnerable devices. Key characteristics include identifying vulnerable assets, triaging and applying patches based on a priority list, and tracking and monitoring the state of those assets.

3.4.5 Scenario 4: Emergency workaround (and backout if needed)

This is the emergency procedure in a crisis situation to temporarily mitigate risk for vulnerabilities prior to a vendor releasing a patch. It is typically required when the vulnerability is being actively exploited in the wild. The workaround can vary and may or may not need to be rolled back afterward. The scope of targets is the same as scenario 1. Organizations need to be prepared to quickly implement a wide variety of emergency workarounds to protect vulnerable devices. Without processes, procedures, and tools in place to implement workarounds, too much time may be lost and vulnerable devices may be compromised before workarounds are in place. This may require disabling system functionality, having automated mechanisms to apply these changes, and having capabilities to revert back these changes when a permanent and approved patch is released.

3.4.6 Scenario 5: Isolation of unpatchable assets

This is the reference architecture and implementation of isolation methods to mitigate the risk of systems which cannot be easily patched. This is typically required if routine patching is not able to accommodate these systems within a reasonable timeframe (usually X days or less). Most systems in this scope are legacy unsupported systems or systems with very high operational uptime requirements.

Isolation is a form of workaround that can be highly effective at stopping threats against vulnerable devices. Organizations need to be prepared to implement isolation methods when needed and to undo the isolation at the appropriate time to restore regular device access and functionality.

3.4.7 Scenario 6: Patch management system security (or other system with administrative privileged access)

This is a reference architecture and implementation of recommended security practices for systems like patch management systems which have administrative privileged access over many other systems. This will include practices like least privilege, privileged access workstations, and software updates.

3.5 Risk Assessment

NIST SP 800-30 Revision 1, *Guide for Conducting Risk Assessments*, states that risk is "a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence." The guide further defines risk assessment as "the process of identifying, estimating, and prioritizing risks to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, other organizations, and the Nation, resulting from the operation of an information system. Part of risk management incorporates threat and vulnerability analyses, and considers mitigations provided by security controls planned or in place."

The NCCoE recommends that any discussion of risk management, particularly at the enterprise level, begins with a comprehensive review of NIST SP 800-37 Revision 2, *Risk Management Framework for Information Systems and Organizations*—material that is available to the public. The Risk Management Framework (RMF) guidance, as a whole, proved to be invaluable in giving us a baseline to assess risks, from which we developed the project, the security characteristics of the build, and this guide. Also, the NIST Cybersecurity Framework and NIST SP 800-53 Revision 5, *Security and Privacy Controls for Information Systems and Organizations* informed our risk assessment and subsequent recommendations from which we developed the security characteristics of the build and this guide.

3.5.1 Threats, Vulnerabilities, and Risks

The objective of this project is to demonstrate example solutions for each of the scenarios described in Section 3.4. Scenarios 0 through 5 collectively address improving the mitigation of software vulnerabilities in small to large IT enterprises for general IT assets. The last scenario, Scenario 6 (see Section 3.4.7) focuses on the security of the patch management technologies themselves. Scenario 6 has a different set of threats, vulnerabilities, and risks than the other scenarios, so it is discussed separately in this section.

Scenarios 0 through 5

Collectively, the objective of Scenarios 0 through 5 is to ensure that software vulnerabilities are mitigated, either through patching or by using additional security controls, for firmware, operating systems, applications, and any other forms of software. The pertinent threats encompass the enormous range of attackers and attacks that target software vulnerabilities. Major risks can be grouped into three categories:

- **Vulnerabilities aren't mitigated, leaving them susceptible to compromise.** Potential causes of this include organizations being unaware of vulnerabilities or vulnerable assets, patching being delayed because of limited resources, users declining to install patches or reboot devices in order for patches to take effect, and organizations choosing not to implement workarounds or isolation techniques to protect unpatchable assets.
- **Installing patches causes unintended side effects.** Examples include breaking the patched software or other software on the asset, inadvertently altering configuration settings to weaken security, adding software functionality without adequately securing that functionality, and disrupting interoperability with other software or assets.
- **Patch integrity is compromised.** A patch's integrity could be compromised at several places in the path from vendor to asset. Examples include the software vendor itself being compromised, the organization downloading patches from an unauthorized source, patches being tampered with while in transit to the organization, and patches being altered in storage at the organization.

Scenario 6

The objective of Scenario 6 is to protect the example solution itself from compromise. To be effective, the example solution requires administrative privileged access for many assets, so this makes it an attractive target for attackers. The example solution also holds sensitive information regarding what computing assets the organization has and what vulnerabilities each asset has, so safeguarding this information from attackers is important. Vulnerabilities that the example solution might have include software vulnerabilities in its own components, misconfigurations, and security design errors, such as not encrypting its network communications.

3.5.2 Security Control Map

Table 3-1 provides a security mapping for Scenarios 0 through 5. It maps the characteristics of the commercial products that the NCCoE has applied to the applicable standards and best practices described in the Framework for Improving Critical Infrastructure Cybersecurity (Cybersecurity Framework) [1] and NIST SP 800-53 Revision 5. This exercise is meant to demonstrate the real-world applicability of standards and recommended practices, but does not imply that products with these characteristics will meet your industry's requirements for regulatory approval or accreditation.

446 **Table 3-1: Mapping Security Characteristics of the Example Solution for Scenarios 0-5**

Cybersecurity Framework Category	Cybersecurity Framework Subcategory	SP 800-53 Revision 5 Controls
Asset Management (ID.AM): The data, personnel, devices, systems, and facilities that enable the organization to achieve business purposes are identified and managed consistent with their relative importance to organizational objectives and the organization's risk strategy.	**ID.AM-1:** Physical devices and systems within the organization are inventoried	**CM-8**, System Component Inventory
	ID.AM-2: Software platforms and applications within the organization are inventoried	**CM-8**, System Component Inventory
Identity Management, Authentication and Access Control (PR.AC): Access to physical and logical assets and associated facilities is limited to authorized users, processes, and devices, and is managed consistent with the assessed risk of unauthorized access to authorized activities and transactions.	**PR.AC-4:** Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties	**AC-3**, Access Enforcement
	PR.AC-5: Network integrity is protected (e.g., network segregation, network segmentation)	**AC-3**, Access Enforcement
Data Security (PR.DS): Information and records (data) are managed consistent with the organization's risk strategy to protect the confidentiality, integrity, and availability of information.	**PR.DS-6:** Integrity checking mechanisms are used to verify software, firmware, and information integrity	**SI-7**, Software, Firmware, and Information Integrity

Cybersecurity Framework Category	Cybersecurity Framework Subcategory	SP 800-53 Revision 5 Controls
Information Protection Processes and Procedures (PR.IP): Security policies (that address purpose, scope, roles, responsibilities, management commitment, and coordination among organizational entities), processes, and procedures are maintained and used to manage protection of information systems and assets.	**PR.IP-3:** Configuration change control processes are in place	**CM-3**, Configuration Change Control
	PR.IP-12: A vulnerability management plan is developed and implemented	**RA-3**, Risk Assessment **RA-5**, Vulnerability Monitoring and Scanning **RA-7**, Risk Response **SI-2**, Flaw Remediation
Protective Technology (PR.PT): Technical security solutions are managed to ensure the security and resilience of systems and assets, consistent with related policies, procedures, and agreements.	**PR.PT-1:** Audit/log records are determined, documented, implemented, and reviewed in accordance with policy	**AU-6**, Audit Record Review, Analysis, and Reporting
Security Continuous Monitoring (DE.CM): The information system and assets are monitored to identify cybersecurity events and verify the effectiveness of protective measures.	**DE.CM-1:** The network is monitored to detect potential cybersecurity events	**CA-7,** Continuous Monitoring
	DE.CM-8: Vulnerability scans are performed	**RA-3**, Risk Assessment **SI-4**, System Monitoring

Table 3-2 provides a security mapping for Scenario 6. Although it has the same format as Table 3-1, the two tables have different functions. Table 3-1 lists the Cybersecurity Framework Subcategories and SP 800-53 Revision 5 security controls that the example solution supports. Table 3-2 lists the Cybersecurity Framework Subcategories and SP 800-53 Revision 5 security controls that are needed to support the example solution—to mitigate the risks of the solution itself.

452 **Table 3-2: Mapping Security Characteristics of the Example Solution for Scenario 6**

Cybersecurity Framework Category	Cybersecurity Framework Subcategory	SP 800-53 Revision 5 Controls
Identity Management, Authentication and Access Control (PR.AC): Access to physical and logical assets and associated facilities is limited to authorized users, processes, and devices, and is managed consistent with the assessed risk of unauthorized access to authorized activities and transactions.	**PR.AC-4:** Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties	**AC-3**, Access Enforcement **AC-5**, Separation of Duties **AC-6**, Least Privilege
	PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi- factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks)	**AC-2**, Account Management **IA-2**, Identification and Authentication (Organizational Users) **IA-3**, Device Identification and Authentication **IA-4**, Identifier Management **IA-5**, Authenticator Management **IA-9**, Service Identification and Authentication
Data Security (PR.DS): Information and records (data) are managed consistent with the organization's risk strategy to protect the confidentiality, integrity, and availability of information.	**PR.DS-1:** Data-at-rest is protected	**SC-28**, Protection of Information at Rest
	PR.DS-2: Data-in-transit is protected	**SC-8**, Transmission Confidentiality and Integrity
Protective Technology (PR.PT): Technical security solutions are managed to ensure the security and resilience of systems and assets, consistent with related policies, procedures, and agreements.	**PR.PT-3:** The principle of least functionality is incorporated by configuring systems to provide only essential capabilities	**CM-7**, Least Functionality

4 Components of the Example Solution

This section highlights the components of the example solution and the collaborators who contributed those components and participated in the solution design, implementation, configuration, troubleshooting, and/or testing. More information on each component, including instructions for installing and configuring it as part of the example solution, is provided in NIST SP 1800-31C, How-To Guides.

4.1 Collaborators

Collaborators that participated in this build and the capabilities of their contributions to the example solution are described briefly in the subsections below.

4.1.1 Cisco

Cisco Systems is a provider of enterprise, telecommunications, and industrial networking solutions. Cisco Identity Services Engine (ISE) enables a dynamic and automated approach to policy enforcement that simplifies the delivery of highly secure, micro-segmented network access control. ISE empowers software-defined access and automates network segmentation within IT and OT environments. Cisco Firepower Threat Defense (FTD) is a threat-focused, next-generation firewall with unified management. It provides advanced threat protection before, during, and after attacks. By delivering comprehensive, unified policy management of firewall functions, application control, threat prevention, and advanced malware protection from the network to the endpoint, it increases visibility and security posture while reducing risks. Learn more about Cisco Systems at https://www.cisco.com.

4.1.2 Eclypsium

Eclypsium is the enterprise firmware security company. The cloud-based solution identifies, verifies, and fortifies firmware and hardware in laptops, servers, network gear, and devices. Eclypsium Administration and Analytics Service secures against persistent and stealthy firmware attacks, provides continuous device integrity, delivers firmware patching at scale, and prevents ransomware and malicious implants. Eclypsium also provides an on-premises version that has parity with the cloud-based platform.

4.1.3 Forescout

Forescout assesses device security posture in real time upon connection and initiates remediation workflows with your existing security tools to enforce compliance. It continuously monitors all devices for new threats and reassesses their patch level hygiene every time the device leaves and returns to the corporate network. Forescout works to assess all device types, including transient devices often missed by point-in-time scans, without requiring agents. Forescout's solution goes beyond simple device authentication to identify every device, assess its security posture, trigger remediation workflows, and

implement access control across heterogeneous networks to unpatched assets. It continuously monitors all connected devices and automates response when noncompliance or unpatched assets are detected.

4.1.4 IBM

IBM MaaS360 with Watson is a unified endpoint management (UEM) solution that transforms how organizations support users, apps, content, and data across every type of mobile device: laptops, smartphones, tablets, and IoT. IBM MaaS360 was built almost twenty years ago as a cloud-based Software-as-a-Service (SaaS) platform that integrates with preferred security and productivity tools, allowing modern business leaders to derive immediate value. IBM MaaS360 is the only UEM platform that leverages the power of the Watson Artificial Intelligence engine to deliver contextually relevant security insights for administrators, while ensuring continuous monitoring of the riskiest end users.

IBM Code Risk Analyzer was developed in conjunction with IBM Research projects and customer feedback. It enables developers to quickly assess and remediate security and legal risks that they are potentially introducing into their source code, and it provides feedback directly in Git artifacts (for example, pull/merge requests) as part of continuous delivery in a DevOps pipeline. IBM Code Risk Analyzer is provided as a set of Tekton tasks, which can be easily incorporated into delivery pipelines.

4.1.5 Lookout

Lookout is an integrated endpoint-to-cloud security solution provider with mobile endpoint protection offerings. Lookout's Mobile Endpoint Security (MES) solution provides cloud-centric behavior-based detection capabilities; it performs behavioral analysis based on telemetry data from nearly 200 million devices and over 120 million apps. This analysis enables Lookout to deliver efficient protection with a lightweight app on the device that optimizes processor speed and battery life. In addition, continuously monitoring changes to the endpoint enables detection of risks that span from jailbreaking or rooting a device to advanced device compromise. With insight into both real-time changes on a device and the aggregate view of behavior across the broader mobile ecosystem, Lookout endpoint protection can detect zero-day threats.

4.1.6 Microsoft

Microsoft Endpoint Manager helps deliver the modern workplace and modern management to keep your data secure in the cloud and on-premises. Endpoint Manager includes the services and tools you use to manage and monitor mobile devices, desktop computers, virtual machines, embedded devices, and servers. Endpoint Manager combines several services, including Configuration Manager (Microsoft Endpoint Configuration Manager), an on-premises management solution for desktops, servers, and laptops that are on your network or internet-based. Endpoint Configuration Manager can be integrated with Intune, Azure Active Directory (AD), Microsoft Defender for Endpoint, and other cloud services. Endpoint Configuration Manager can deploy apps, software updates, and operating systems, and also be used to monitor compliance and to query and act on clients in real time.

4.1.7 Tenable

Tenable.sc is Tenable's on-premises vulnerability management solution. Built on Nessus technology, the Tenable.sc family of products identifies, investigates, and prioritizes vulnerabilities. You get real-time, continuous assessment of your security and compliance posture so you can discover unknown assets and vulnerabilities, monitor unexpected network changes, and prioritize weaknesses to minimize your cyber risk and prevent breaches. Tenable.sc includes over 350 pre-built, highly customizable dashboards and reports to give you immediate insight into your security compliance, effectiveness, and risk. You can continuously measure, analyze, and visualize the effectiveness of your security program, based on high-level business objectives and underlying customizable policies that executives care about.

Powered by Nessus technology and managed in the cloud, Tenable.io provides the industry's most comprehensive vulnerability coverage with the ability to predict which security issues to remediate first. Using an advanced asset identification algorithm, Tenable.io provides the most accurate information about dynamic assets and vulnerabilities in ever-changing environments. As a cloud-delivered solution, its intuitive dashboard visualizations, comprehensive risk-based prioritization, and seamless integration with third-party solutions help security teams maximize efficiency and scale for greater productivity.

4.1.8 VMware

VMware vRealize Automation includes SaltStack Config, a modern configuration management platform with the performance, speed, and agility IT teams need to manage large, complex IT systems and improve efficiency at scale. For this project, vRealize Automation SaltStack Config provides device configuration and software distribution capabilities. Specifically, it allows for configuration changes to be made to devices by updating or removing software as well as updating settings such as network information.

SaltStack SecOps, an add-on to the vRealize products, gives system administrators the ability to create security policies and scan assets to determine whether they are compliant with supported, industry-recognized security benchmarks. SaltStack SecOps also has the ability to scan your system for Common Vulnerabilities and Exposures (CVEs), then immediately apply the updates or patches to remediate the advisories.

4.2 Technologies

Table 4-1 lists all the technologies used in this project, the primary functions that each technology provides to the project, and the Cybersecurity Framework Subcategories that the technology supports in this project. Please refer to Table 3-1 for an explanation of the NIST Cybersecurity Framework Subcategory codes.

553 **Table 4-1: Technologies Used in the Build**

Technology	Primary Functions	Cybersecurity Framework Subcategories
Cisco Firepower Threat Defense (FTD) and Cisco Firepower Management Center (FMC)	Network policy enforcement	PR.AC-4, PR.AC-5, DE.CM-1
Cisco Identity Services Engine (ISE)	Asset discovery and inventory; network access control	ID.AM-2, PR.AC-4, PR.AC-5
Eclypsium Administration and Analytics Service	Hardware and firmware inventory; firmware vulnerability assessment, integrity monitoring, and updating	ID.AM-1, ID.AM-2, PR.DS-6, PR.IP-12
Forescout Platform	Asset discovery and inventory; security policy enforcement	ID.AM-2, PR.AC-4, PR.AC-5, PR.IP-3, PR.PT-1
IBM Code Risk Analyzer	Vulnerability scanning for source code	PR.IP-12
IBM MaaS360 with Watson	Asset inventory; configuration management; software updates	ID.AM-2, PR.IP-3, PR.IP-12
Lookout Mobile Endpoint Security (MES)	Security policy enforcement; vulnerability scanning and reporting; software discovery and inventory; firmware vulnerability assessment and policy enforcement	PR.AC-4, PR.IP-3, PR.IP-12
Microsoft Endpoint Configuration Manager	Asset discovery; configuration management; software updates	ID.AM-2, PR.IP-3, PR.IP-12
Tenable.sc, Tenable.io, and Nessus	Asset discovery and inventory; vulnerability scanning and reporting	ID.AM-2, PR.PT-1, DE.CM-8
VMware vRealize Automation SaltStack Config and SaltStack SecOps	Vulnerability scanning and remediation; configuration management; software updates	PR.IP-3, PR.IP-12, DE.CM-8

554 The following sections summarize the security capabilities that each technology provided to the
555 example solution.

4.2.1 Cisco Firepower Threat Defense (FTD) & Firepower Management Center (FMC)

Cisco Firepower Threat Defense (FTD) is a virtual firewall that was utilized as the networking backbone that connected all of the lab subnets. This build also used the Cisco FTD firewall to provide network access management capabilities, including enforcing network access control using firewall rules. Cisco FTD was deployed and managed in the lab via a separate Cisco Firepower Management Center (FMC) virtual machine.

To support the unpatchable asset scenario (Scenario 5), the integration between Cisco FTD and Cisco Identity Services Engine (ISE) via pxGrid allowed for the firewall to ingest security group tags (SGTs) that were applied by ISE. SGTs were then used in custom firewall rules to restrict network access to any machine that was given a quarantine tag. Section 4.2.2 has more information on this integration.

4.2.2 Cisco Identity Services Engine (ISE)

In this build Cisco Identity Services Engine (ISE) provided asset discovery and inventory, and network access control to enforce administrator-created security and access control policies. Cisco ISE had integrations with several other example solution technologies, including the following:

- An integration between ISE and AD allowed the user of a device to be identified. This information could then be used in custom policy.
- A Dynamic Host Configuration Protocol (DHCP) relay was established between ISE and the lab DHCP server. This integration allowed for ISE to identify any device that was assigned an IP address. This allowed devices to be discovered as they joined the network.
- Cisco ISE was configured to integrate with Tenable.sc via an adapter. Cisco ISE leveraged this adapter to prompt Tenable to scan devices newly connected to the network. Cisco ISE could then ingest this scan data to find the Common Vulnerability Scoring System (CVSS) scores of device vulnerabilities. An ISE policy was written to apply a quarantine action, via SGTs, to any device with a CVSS score equal to or greater than 7 (corresponding to high and critical vulnerabilities).
- Cisco Platform Exchange Grid (pxGrid) was configured to share contextual information about authenticated devices to the firewall. Cisco ISE was utilized to apply SGTs to devices as they were assessed for vulnerabilities. These SGTs were then passed to the lab firewall via pxGrid, where they could be used in custom firewall rules. PxGrid was also used to share communications between Forescout and Cisco ISE. Forescout could apply a quarantine tag to observed devices, which would then be shared with ISE.

4.2.3 Eclypsium Administration and Analytics Service

In this build, we utilized Eclypsium Administration and Analytics Service to provide agent-based identification of hardware and firmware for our laptop, desktop, and server endpoints while also

monitoring the firmware for vulnerable or end-of-life versions. Eclypsium monitored laptop and virtual machine (VM) firmware integrity, and alerted if a component or its associated firmware has changed. It also monitored endpoints for known security vulnerabilities from out-of-date firmware. Finally, we utilized Eclypsium's beta firmware update script, which automatically finds the latest known Basic Input/Output System (BIOS) firmware version for the system, downloads the update, and executes it to update the BIOS.

4.2.4 Forescout Platform

In this build the Forescout platform was configured to perform endpoint discovery by detecting endpoints and determining software information about those endpoints based on a set of attributes. Forescout also provided the capability to isolate or restrict assets that cannot be patched and to respond to emergency scenarios, such as providing a workaround or deploying an emergency patch. Forescout had several integrations with other example solution technologies:

- The User Directory plugin was configured so that the Forescout platform integrated with the lab's AD Domain Controller. This plugin provided Lightweight Directory Access Protocol (LDAP) services to Forescout, allowing directory-based users to log in into Forescout as well as providing user directory information such as the current active domain users logged into each endpoint.
- The Domain Name System (DNS) Query Extension configuration setting allowed Forescout to query the DNS server to determine the hostnames of devices identified by Forescout.
- The Tenable VM plugin provided the Forescout platform with vulnerability and scan status information which can be used to create custom policies. This plugin also enabled Forescout to utilize vulnerability management information that Tenable.sc collected from endpoints, and allowed Forescout to determine if scans had been performed on endpoints within the lab.
- The Microsoft Systems Management Server (SMS)/System Center Configuration Manager (SCCM) module was configured to allow the Forescout platform to integrate with Microsoft Endpoint Configuration Manager. This module allowed for a custom policy to be created that used data from Microsoft Endpoint Configuration Manager.
- The Linux plugin was configured to collect information from and manage Linux-based endpoints via two methods: secure shell (SSH) access to the endpoint, and agent-based integration with the endpoint.
- The HPS Inspection Engine was configured to collect information from Windows endpoints via two methods. The first method utilized a directory-based integration with the lab's AD Domain Services instance, which collected domain-based information on the Windows endpoint. The second method utilized an agent-based integration called SecureConnector that allowed Forescout to collect and manage Windows endpoints.
- The pxGrid plugin was configured to integrate with Cisco ISE. This plugin gave the Forescout platform the ability to utilize Cisco ISE to apply adaptive network control (ANC) policies to endpoints for restricting their network access.

- The Switch plugin was configured to integrate Forescout with the physical Cisco switch located in the lab. The plugin used information from the switch to collect information about endpoints that were physically connected to the switch.

Our implementation utilized multiple policies to support the use case scenarios. Examples of capabilities that the policies provided are described below:

- Check for a particular application running on Windows; if present, stop execution and uninstall it.
- Check an endpoint for known critical vulnerabilities; if any are present, use Cisco ISE to quarantine the endpoint via the pxGrid plugin.
- Force a Windows update to occur on an endpoint with Windows Update enabled.
- Determine if a Windows endpoint has the Microsoft Endpoint Configuration Manager agent installed.

4.2.5 IBM Code Risk Analyzer

IBM Code Risk Analyzer was used to demonstrate vulnerability scanning and reporting for pre-deployed code as part of a DevOps pipeline to deliver a cloud-native application. Integration with Git allowed the Code Risk Analyzer to perform vulnerability assessments against applications and base images. The Code Risk Analyzer would then print a bill-of-materials, which indicates the composition of a deployment. This allows an administrator to see all of an application's dependencies and their sources, providing visibility into application components which could have vulnerabilities.

4.2.6 IBM MaaS360 with Watson

IBM MaaS360 with Watson was used to demonstrate how to securely manage an enterprise's devices by enabling deployment, control of content, and policy controls. Enterprises can manage organization-owned and user-owned devices using this product. The lab used MaaS360 for asset identification and assessment, routine patching and emergency patching, emergency workarounds, and isolation of assets that cannot be patched. The first phase of this lab build used MaaS360's comprehensive enterprise mobility management (EMM) capability to manage a MacBook Pro and a Windows 10 virtual desktop. The second phase used MaaS360's Mobile Device Manager (MDM) capability to manage Android and Apple iOS devices.

This build also used Maas360's Cloud Extender, which allows enterprises to integrate mobile devices with corporate on-premises and cloud-based resources. The Cloud Extender was installed on the AD server to allow users to log in with AD accounts.

4.2.7 Lookout

Lookout MES was used in this build to perform security compliance, vulnerability scanning, and firmware/software discovery for mobile endpoints. Our implementation of Lookout MES was integrated with IBM MaaS360. Lookout MES shared custom device attributes, such as device threat, with MaaS360, which could in turn provide policy enforcement. The Lookout for Work mobile client was able to provide firmware and application vulnerability assessment for mobile endpoints. Administrators could use Lookout to see which vulnerabilities were affecting deployed endpoints and find risk grades (i.e., A, B, C, D, or F) for installed applications.

4.2.8 Microsoft Endpoint Configuration Manager

Microsoft Endpoint Configuration Manager was used in this build to perform configuration management, including software and firmware patching, for Windows-based hosts. Our implementation of Endpoint Configuration Manager included Windows Server Update Services (WSUS), an update service primarily used for downloading, distributing, and managing updates for Microsoft Windows-based systems. The example build used Microsoft Endpoint Configuration Manager to demonstrate the identification of endpoints utilizing Heartbeat discovery and Windows Domain discovery methods, the patching of Windows endpoints via Microsoft updates and third-party update sources, and the deployment of custom scripts to endpoints.

4.2.9 Tenable.io

In the example build, Tenable.io was used to provide vulnerability scanning and reporting for Docker container images. Containers are built from images and vulnerabilities are patched in images, not deployed containers, so images are the focus of scanning. Tenable.io scanned the repository of a Red Hat OpenShift cluster in the lab environment. Tenable.io was scheduled to routinely pull the latest images from the OpenShift cluster and perform vulnerability scans on them. Scan information was reported in the container security section of the Tenable.io Web Console. Administrators could see vulnerability information for containers deployed in their respective networks.

4.2.10 Tenable.sc and Nessus

This example build utilized two Tenable products in the first phase of this project, Nessus and Tenable.sc. We used Nessus to scan Linux, Windows, and macOS endpoints and network switches for vulnerability data, and then feed this information to Tenable.sc for reporting. Tenable.sc, a vulnerability management product, collected the information from Nessus and reported that information to administrators using dashboards and reports. Also, Tenable.sc had integrations with other example solution technologies:

- An integration between Tenable.sc and Cisco ISE was performed to initiate scans of any newly connected network devices. Tenable.sc would pass scan data to Cisco ISE, where a custom policy was written to quarantine devices based on their CVSS scores.
- An integration between Forescout and Tenable was leveraged to scan devices as hosts joined the network. Forescout could prompt Tenable to scan hosts to determine if an endpoint had critical vulnerabilities. This information was ingested by Forescout for the purpose of quarantining endpoints.

4.2.11 VMware vRealize Automation SaltStack Config

In this example build, vRealize Automation SaltStack Config was used to provide configuration management and patch deployment. In the first phase of the build, it was used to manage Windows workstations and servers, a macOS laptop, and Linux/Unix-based VMs and servers. SaltStack Config was configured to run jobs, applying different states or configurations, on endpoints. The job that was written for this project, in support of the emergency workaround scenario, could uninstall an application based on the current version of the product. SaltStack Config also had an add-on component called SaltStack SecOps which was utilized to scan devices for known vulnerabilities and provide mitigation actions, including missing updates for endpoints.

Appendix A Patch Management System Security Practices

Section 3.4.7 describes Scenario 6, "Patch management system security (or other system with administrative privileged access)." In support of Scenario 6, this appendix describes recommended security practices for systems like patch management systems which have administrative privileged access over many other systems as defined as "critical software" in Executive Order (EO) 14028. It then summarizes how the example solution components described in this practice guide could support each of those recommended security practices.

A.1 Security Measures

The table below defines security measures for software of critical importance. Note that these security measures are not intended to be comprehensive. They are based on those in the NIST publication *Security Measures for "EO-Critical Software" Use Under Executive Order (EO) 14028*. A *security measure (SM)* is a high-level security outcome statement that is intended to apply to critical software or to all platforms, users, administrators, data, or networks (as specified) that are part of running critical software. The security measures are grouped by five objectives:

1. Protect critical software and *critical software platforms* (the platforms on which critical software runs, such as endpoints, servers, and cloud resources) from unauthorized access and usage.
2. Protect the confidentiality, integrity, and availability of data used by critical software and critical software platforms.
3. Identify and maintain critical software platforms and the software deployed to those platforms to protect the critical software from exploitation.
4. Quickly detect, respond to, and recover from threats and incidents involving critical software and critical software platforms.
5. Strengthen the understanding and performance of humans' actions that foster the security of critical software and critical software platforms.

Each row in the table defines one security measure and lists mappings to it from the NIST Cybersecurity Framework and NIST SP 800-53 Revision 5, Security and Privacy Controls for Information Systems and Organizations. These mappings are in the forms of Cybersecurity Framework Subcategories and SP 800-53 security controls, respectively. The mappings are general and informational; any particular situation might have somewhat different mappings.

Security Measure (SM)	Cybersecurity Framework Subcategories	SP 800-53 Rev. 5 Controls
Objective 1: Protect critical software and critical software platforms from unauthorized access and usage.		
SM 1.1: Use multi-factor authentication that is verifier impersonation-resistant for all users and administrators of critical software and critical software platforms.	PR.AC-1, PR.AC-7	AC-2, IA-2, IA-4, IA-5
SM 1.2: Uniquely identify and authenticate each service attempting to access critical software or critical software platforms.	PR.AC-1, PR.AC-7	AC-2, IA-9
SM 1.3: Follow privileged access management principles for network-based administration of critical software and critical software platforms. Examples of possible implementations include using hardened platforms dedicated to administration and verified before each use, requiring unique identification of each administrator, and proxying and logging all administrative sessions to critical software platforms.	PR.AC-1, PR.AC-7, PR.MA-1, PR.MA-2	AC-2, IA-2, SC-2, SC-7 enhancement 15
SM 1.4: Employ boundary protection techniques as appropriate to minimize direct access to critical software, critical software platforms, and associated data. Examples of such techniques include network segmentation, isolation, software-defined perimeters, and proxies.	PR.AC-3, PR.AC-5	SC-7
Objective 2: Protect the confidentiality, integrity, and availability of data used by critical software and critical software platforms.		
SM 2.1: Establish and maintain a data inventory for critical software and critical software platforms.	ID.AM-3, DE.AE-1	CM-8, PM-5
SM 2.2: Use fine-grained access control for data and resources used by critical software and critical software platforms to enforce the principle of least privilege to the extent possible.	PR.AC-4	AC-2, AC-3, AC-6
SM 2.3: Protect data at rest by encrypting the sensitive data used by critical software and critical software platforms consistent with NIST's cryptographic standards.	PR.DS-1	SC-28
SM 2.4: Protect data in transit by using mutual authentication whenever feasible and by encrypting sensitive data communications for critical software and critical software platforms consistent with NIST's cryptographic standards.	PR.AC-3, PR.AC-7, PR.DS-2, PR.PT-4, DE.CM-7	AC-4, AC-17, SC-8

Security Measure (SM)	Cybersecurity Framework Subcategories	SP 800-53 Rev. 5 Controls
SM 2.5: Back up data, exercise backup restoration, and be prepared to recover data used by critical software and critical software platforms at any time from backups.	PR.IP-4	CP-9, CP-10
Objective 3: Identify and maintain critical software platforms and the software deployed to those platforms to protect the critical software from exploitation.		
SM 3.1: Establish and maintain a software inventory for all platforms running critical software and all software (both critical and non-critical) deployed to each platform.	ID.AM-1, ID.AM-2, ID.SC-2	CM-8, PM-5, RA-9
SM 3.2: Use patch management practices to maintain critical software platforms and all software deployed to those platforms. Practices include: ▪ rapidly identify, document, and mitigate known vulnerabilities (e.g., patching, updating, upgrading software to supported version) to continuously reduce the exposure time ▪ monitor the platforms and software to ensure the mitigations are not removed outside of change control processes	ID.RA-1, ID.RA-2, ID.RA-6, PR.IP-12, DE.CM-8, RS.MI-3	CA-7, RA-5, SI-2, SI-5, SR-8
SM 3.3: Use configuration management practices to maintain critical software platforms and all software deployed to those platforms. Practices include: ▪ identify the proper hardened security configuration for each critical software platform and all software deployed to that platform (hardened security configurations enforce the principles of least privilege, separation of duties, and least functionality) ▪ implement the configurations for the platforms and software ▪ control and monitor the platforms and software to ensure the configuration is not changed outside of change control processes	ID.RA-1, ID.RA-2, ID.RA-6, PR.AC-4, PR.IP-1, PR.IP-3, PR.PT-3, DE.CM-8, RS.MI-3	AC-5, AC-6, CA-7, CM-2, CM-3, CM-6, CM-7, RA-5, SI-5
Objective 4: Quickly detect, respond to, and recover from threats and incidents involving critical software and critical software platforms.		
SM 4.1: Configure logging to record the necessary information about security events involving critical software platforms and all software running on those platforms.	PR.PT-1	AU-2, AU-3, AU-4, AU-5, AU-8, AU-9, AU-11, AU-12

Security Measure (SM)	Cybersecurity Framework Subcategories	SP 800-53 Rev. 5 Controls
SM 4.2: Continuously monitor the security of critical software platforms and all software running on those platforms.	DE.CM-7	CA-7, SI-4
SM 4.3: Employ endpoint security protection on critical software platforms to protect the platforms and all software running on them. Capabilities include: ▪ protecting the software, data, and platform by identifying, reviewing, and minimizing the attack surface and exposure to known threats ▪ permitting only verified software to execute (e.g., file integrity verification, signed executables, allowlisting) ▪ proactively detecting threats and stopping them when possible ▪ responding to and recovering from incidents ▪ providing the necessary information for security operations, threat hunting, incident response, and other security needs	PR.DS-5, PR.DS-6, DE.AE-2, DE.CM-4, DE.CM-7, DE.DP-4	SI-3, SI-4, SI-7
SM 4.4: Employ network security protection to monitor the network traffic to and from critical software platforms to protect the platforms and their software using networks. Capabilities include: ▪ proactively detecting threats at all layers of the stack, including the application layer, and stopping them when possible ▪ providing the necessary information for security operations, threat hunting, incident response, and other security needs	PR.DS-5, DE.AE-1, DE.AE-3, DE.CM-1, DE.CM-4, DE.CM-7, DE.DP-4	AU-13, AU-14, SC-7, SI-3
SM 4.5: Train all security operations personnel and incident response team members, based on their roles and responsibilities, on how to handle incidents involving critical software or critical software platforms.	PR.AT-5, PR.IP-9, PR.IP-10	AT-3, CP-3, IR-2
Objective 5: Strengthen the understanding and performance of humans' actions that foster the security of critical software and critical software platforms.		
SM 5.1: Train all users of critical software, based on their roles and responsibilities, on how to securely use the software and the critical software platforms.	PR.AT-1	AT-2, AT-3
SM 5.2: Train all administrators of critical software and critical software platforms, based on their roles and responsibilities, on how to securely administer the software and/or platforms.	PR.AT-2	AT-3, CP-3

Security Measure (SM)	Cybersecurity Framework Subcategories	SP 800-53 Rev. 5 Controls
SM 5.3: Conduct frequent awareness activities to reinforce the training for all users and administrators of critical software and platforms, and to measure the training's effectiveness for continuous improvement purposes.	PR.AT-1, PR.AT-2	AT-3

A.2 Component Support of Security Measures

This section provides summary tables for how each technology provider's components in the example solution could support the security measures defined above. The technical mechanisms, configuration settings, or other ways in which the components could provide this support were not necessarily utilized in the example solution build. The information is provided here to offer examples of how these security measures might be implemented, not to serve as recommendations for how to implement them.

Each table in this section has the same four columns:

- **SM #**: This lists a security measure ID from the previous section and hyperlinks to the definition of that ID.
- **Question**: This contains a question NIST asked the technology providers to answer for their components regarding the associated security measure.
- **Technical Mechanism or Configuration**: This is a summary of the answer from the component's technology provider. The content submitted by each technology provider has been edited for brevity.
- **Refs.**: This provides hyperlinks to any applicable references specified by the technology provider. This column is blank if no reference was needed or available, or if there is a single reference for all entries in a table, in which case the reference is defined immediately before the table.

In each table, rows with no answer or an answer of "no" or "not applicable" have been omitted for brevity.

A.2.1 Cisco FTD Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	Certificates from a Personal Identity Verification (PIV) card or Common Access Card (CAC) can be used along with soft certificates to authenticate admin users.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	Services using the PxGrid solution to gather data from the system or publish require the use of certificates to secure the communications channel.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	The Cisco FMC admin console supports role-based access control. There are predefined roles, and custom roles with permissions can be created.	Ref1
SM 1.4	Does the system allow for the use of discretionary access control lists (DACLs), network segmentation, or isolation for access to the platform?	Administrators can limit access by IP address and port.	Ref1
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	The Cisco FMC admin console and command-line interface (CLI) both support role-based access control.	Ref1
SM 2.5	Does the system support performing regular backups and restorations?	Cisco FMC enables backup and restore of configuration and monitoring. FMC also provides backup and restore of the devices it manages.	Ref1
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	Cisco distributes several types of upgrades and updates for Firepower deployments. These include OS versions, patches, vulnerability databases, intrusion rules, and geolocation databases. These are all deployed centrally from FMC.	Ref1
SM 4.1	Does the security tool support logging and sending that data to rsyslog or Security Information and Event Management (SIEM)?	FMC allows for sending all logs to a third-party SIEM using syslog or eStreamer.	Ref1
SM 4.4	Does the platform allow for logging connection events to the tool?	The system can generate logs of the connection events its managed devices detect. Connection events include Security Intelligence events (connections blocked by the reputation-based Security Intelligence feature.)	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Cisco provides training resources through direct offering, partner, knowledge partners, and on-demand through Cisco Live.	
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Cisco provides training resources through direct offering, partner, knowledge partners, and on-demand through Cisco Live.	
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Cisco regularly collects metrics from completed user training to make improvements and updates.	

757

758

A.2.2 Cisco ISE Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	Certificates from a PIV or CAC can be used along with soft certificates to authenticate admin users.	Ref1
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	Services using the ISE PxGrid solution to gather data from the system or publish require the use of certificates to secure the communications channel.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	The Cisco ISE admin console and CLI both support role-based access control.	Ref1
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	Both the admin user interface (UI) and CLI can be configured to limit IP access to the system.	Ref1
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	The Cisco ISE admin console and CLI both support role-based access control.	Ref1
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	Cisco ISE can be configured for Federal Information Processing Standards (FIPS) compliance. In this mode, only the protocols listed here are allowed to be used for authentication: EAP-TLS, PEAP, EAP-FAST, EAP-TTLS	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 2.5	Does the system support performing regular backups and restorations?	Cisco ISE backs up to a repository both the configuration and event data. The system provides high-availability (HA) capabilities with redundant service pairs.	Ref1
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	Cisco ISE provides a centralized patching mechanism through the admin node to apply patches to all systems that are a member of the deployment. Patches are rollups, so administrators do not have to install multiple patches. Patches include vulnerability fixes and bug fixes.	Ref1
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	Cisco ISE allows administrators to turn on and off features and functions. Cisco ISE does not allow access to the underlying OS, so services are only enabled and disabled based on the packages needed to support the enabled services.	Ref1
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	Log events for the following categories are sent by all nodes in the deployment to the logging targets: Administrative and Operational Audit, System Diagnostics, and System Statistics.	Ref1
SM 4.4	Does the platform allow for logging connection events to the tool?	The web interface can specify remote syslog server targets to which system log messages are sent. Log messages are sent to the remote syslog server targets in accordance with the syslog protocol standard (RFC 3164).	Ref1
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Cisco provides training resources through direct offering, partner, knowledge partners, and on-demand through Cisco Live.	
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Cisco provides training resources through direct offering, partner, knowledge partners, and on-demand through Cisco Live.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Cisco regularly collects metrics from completed user training to make improvements and updates.	

759

760 A.2.3 Eclypsium Administration and Analytics Service Support of Security
761 Measures

762 All entries in this table have the same two references: the Eclypsium-supplied Solution Guide and
763 Deployment Guide. The Solution Guide is built into the product, and Eclypsium provides the Deployment
764 Guide at purchase, so it was not possible to provide hyperlinks for this table.

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	Eclypsium integrates with multiple authentication mechanisms, many of which support multi-factor authentication (MFA).	
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	Unique application programming interface (API) tokens are managed by Eclypsium administrators.	
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	The Eclypsium platform contains Admin/User access roles. Only administrators can change systemwide analysis policies.	
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	The Linux OS hosting Eclypsium can be configured to allow for the creation of network-based access restrictions.	
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	The Eclypsium platform contains Admin/User access roles. Only administrators can change systemwide analysis policies.	
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	The data-at-rest encryption implementation is done as part of the backend platform onto which Eclypsium is deployed. In the cloud, the provider's key management system may be used. In an on-premises deployment, the OS or hardware-based encryption on the physical servers may be used.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	All communications occur over Transport Layer Security (TLS). FIPS mode can be enabled and utilized where desired.	
SM 2.5	Does the system support performing regular backups and restorations?	Backups of the Eclypsium backend are performed as part of the platform onto which it is deployed. Standard mechanisms for Linux server backup/restore will operate normally.	
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	This information is in the Solution Guide. When scanning firmware on target systems, similar information may be inferred from binary analysis.	
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	The cloud version is managed by Eclypsium to provide updates. The on-premises version is the responsibility of the customer. The OS can be configured to perform updates. On target systems, Eclypsium will indicate whether firmware is up to date.	
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	Eclypsium directly manages the configuration of cloud deployments. In an on-premises environment, configuration management becomes the responsibility of the customer. Normal configuration management for Linux servers will apply to the Eclypsium backend.	
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	In most instances, syslog is integrated with SIEM tools. Eclypsium alerts for target systems are forwarded over syslog to such tools when configured.	
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	There is an audit trail of users who have logged in and the actions they performed. Updates are also sent out to help remediate software running on the platform.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 4.3	Is anti-malware or antivirus able to be installed on the system running your platform?	Eclypsium scanners and the Eclypsium backend are compatible with running other endpoint security software on the same device.	
SM 4.4	Does the platform allow for logging connection events to the tool?	In cloud deployments, Eclypsium manages network security protections. In an on-premises deployment, this would be inherited from the environment into which Eclypsium is deployed.	
SM 4.5	Are there training courses or procedures in the event of an incident involving the tool or platform?	Eclypsium security operations personnel receive security and incident response training. Customer training is available from Eclypsium to cover firmware security and incident response scenarios.	
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Eclypsium has the latest training catalog.	
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Eclypsium has the latest training catalog.	
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Eclypsium has the latest training catalog.	

765

766 A.2.4 Forescout Platform Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	The Forescout platform's integration with PIV and Homeland Security Presidential Directive 12 (HSPD-12) cards allows for this capability.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	The Forescout platform supports a range of accounts with different access levels as required to support least privilege.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	Forescout supports the use of DACLs, virtual local area network (VLAN) assignment, and any other network-based control offered by the network devices in use for device isolation as needed.	Ref1
SM 2.1	Does the software list and maintain an inventory of all software criticalities and integrations?	This is enabled via Forescout's native policy.	Ref1
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	The Forescout platform supports a range of accounts with different access levels as required to support least privilege.	Ref1
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	Forescout natively encrypts the data at rest on the hard drives but can also verify and establish the encryption level of managed endpoints.	Ref1
SM 2.5	Does the system support performing regular backups and restorations?	Forescout supports backup/restore of data and configurations of all appliances.	Ref1
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	Forescout can identify applications and services that are installed and/or running on Windows, Linux, and macOS. Remote inspection capabilities are enabled either by integration with AD (LDAP) or via an agent (Secure Connector). This in turn can be enhanced by creating Forescout security policies to identify all software with enhanced privileges and known CVEs.	
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	Forescout integrates with a variety of patch and OS management tools. Forescout has native remediations via scripting on endpoints via policy.	Ref1
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	Forescout can perform control actions against any managed endpoint. Services as a property are an attribute detected running/installed on the endpoint. These attributes (services) can in turn can be stopped/started or removed as required via policy.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	The Forescout platform will send rich device context information to a SIEM system for logging and event analysis.	Ref1 Ref2
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	Forescout supports a default Windows Vulnerability CVE/Patch plugin (published by Microsoft) to actively scan all Windows clients/servers in real time via policy. The Forescout platform also provides Security Policy Templates (SPT) covering zero-day information and assesses software and hardware for these issues. SPT includes vulnerability and response templates with relevant data for vulnerabilities as documented by Forescout security labs.	Ref1
SM 4.4	Does the platform allow for logging connection events to the tool?	All successful and failed connections to the Forescout platform are logged in system event logs. Administrators can view these logs. An option is also available to forward event messages to third-party logging systems via syslog.	Ref1
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Forescout offers training and certifications for administrators.	Ref1
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Forescout offers training and certifications for engineers.	Ref1

767

768 A.2.5 IBM Code Risk Analyzer Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	It leverages the IBM Cloud authentication mechanism, which provides multi-factor authentication for all users and administrators.	
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	All users and machines are identified using the Identity and Access Management feature of IBM Cloud.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	Accounts can be created and assigned to appropriate roles that have different access levels. This functionality is provided by the Identify and Access Management feature of IBM Cloud.	
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	Network segmentation and isolation is done by using Kubernetes clusters and Istio as the service mesh. Strict policies exist for egress and ingress.	
SM 2.1	Does the software list and maintain an inventory of all software criticalities and integrations?	The software keeps a bill of materials for each component. This bill of materials contains a full list of third-party dependencies. Integration is allowed with only IBM-authorized software.	
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	This feature is achieved by using the Identity and Access Management (IAM) feature of IBM Cloud. IAM has comprehensive features for granular access for users, administrators, and machines.	
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	All data at rest, whether in databases or file systems, is encrypted using NIST-certified cryptographic standards.	
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	All data in transit is encrypted using NIST-certified cryptographic standards. This includes data that is flowing between microservices inside a cluster.	
SM 2.5	Does the system support performing regular backups and restorations?	The system data is backed up regularly for offsite storage. Disaster recovery procedures are reviewed and tested regularly by IBM engineers.	
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	A bill of materials is created for each microservice. Integrations with databases and other systems are tracked. Change management is rigorously followed.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	The OS, middleware, and application components are regularly patched using automated pipelines. These components are scanned for any vulnerabilities and patches are deployed within strict timeframes.	
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	The system is configured and deployed using various standard techniques such as Kubernetes Helm charts and YAML files. The service can be disabled in all regions within minutes by disabling DNS entries, reverse proxies, etc.	
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	Syslog data is streamed to centralized logging mechanisms. The security events data is also made available to clients using the Activity Tracker mechanism.	
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	Continuous monitoring for security is accomplished by using firewalls and service mesh.	
SM 4.3	Is anti-malware or antivirus able to be installed on the system running your platform?	All systems running the system have anti-malware software running on them. Comprehensive reports are generated to ensure compliance.	
SM 4.4	Does the platform allow for logging connection events to the tool?	All successful and unsuccessful connections are logged in the Activity Tracker and in the Identity and Access Management system of IBM Cloud.	
SM 4.5	Are there training courses or procedures in the event of an incident involving the tool or platform?	Process documentation, runbooks, training, and technology are in place to respond to incidents in a timely manner. High-severity incidents are tracked at executive levels. Root-cause analysis is performed and actionable tasks are documented. Best practices are shared across all teams in IBM Cloud.	
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Self-service tutorials are available to users based on their roles. Comprehensive documentation is available as well.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	IBM Garage teams host courses for all aspects of the IBM Cloud platform.	
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Regular trainings are conducted for all developers and administrators that are responsible for operating the IBM Cloud. The training materials are revised as new best practices become available.	

769

770 A.2.6 IBM MaaS360 with Watson Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	Connections to IBM MaaS360 are authenticated with API keys or credentials.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	In the MaaS360 admin console, roles can be assigned to each administrator based on their individual needs.	Ref1
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	In the MaaS360 admin console, custom roles can be defined with granular access rights.	Ref1
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	IBM MaaS360 offers training courses that are catered to the role an individual will hold utilizing the product.	Ref1
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	IBM MaaS360 offers training courses for administrative users.	Ref1 Ref2
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Release Notes are regularly updated with new and updated feature information, and the “MaaS360 Latest” panel provides videos and tutorials on new and updated capabilities. Each training course has a star rating system for effectiveness and improvement purposes.	Ref1

771

A.2.7 Lookout MES Support of Security Measures

SM #	Description	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	Organizations can integrate their existing Security Assertion Markup Language (SAML) 2.0 MFA solutions for authorization purposes into the Lookout MES Console.	
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	Lookout identifies and authenticates each user or machine account that attempts to access the platform. Audit logs also collect actions taken by each account.	
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	Lookout allows for the creation of several administrative types with decreasing levels of access.	
SM 2.1	Does the software list and maintain an inventory of all software criticalities and integrations?	The Lookout MES Console provides a full application inventory list of all devices within the customer's user fleet.	
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	Lookout allows for the creation of several administrative types with decreasing levels of access.	
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	Data at rest is encrypted using Advanced Encryption Standard (AES) 256-bit encryption.	
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	Data in transit is encrypted using TLS version 1.2.	
SM 2.5	Does the system support performing regular backups and restorations?	Daily backups and snapshots of the production environment are taken and stored via Amazon's S3 service within multiple zones and US regions. Regular integrity checks occur through restorations occurring multiple times annually. These restores populate new production instances which are then verified and monitored.	

SM #	Description	Technical Mechanism or Configuration	Refs.
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	The Lookout MES Console provides a full application inventory list of all devices within the customer's user fleet.	
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	Patches to the Lookout MES Console are controlled and maintained by Lookout backoffice support.	
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	Lookout uses a representational state transfer (REST) API to capture and send all console-related logs (e.g., device changes, threat information, system audit events) to SIEMs and syslog readers.	
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	Lookout is Federal Risk and Authorization Management Program (FedRAMP) Moderate and therefore follows strict patch management controls for patching our own software.	
SM 4.4	Does the platform allow for logging connection events to the tool?	Lookout captures connection events to the tool and activities conducted within the tool via our auditing capabilities.	
SM 4.5	Are there training courses or procedures in the event of an incident involving the tool or platform?	Internally, Lookout has established procedures for how to respond to a security incident (leak, compromise, etc.). These procedures follow strict FedRAMP Moderate policies.	
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Lookout provides first-touch training and guidance for using the Lookout MES and for integration guidance with a customer's MDM. Additionally, frequently asked questions (FAQs), integration guides, and console user guides are available to all administrators via the Lookout Support Knowledge portal.	

SM #	Description	Technical Mechanism or Configuration	Refs.
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Lookout provides first-touch training and guidance for using the Lookout MES and for integration guidance with a customer's MDM. Additionally, FAQs, integration guides, and console user guides are available to all administrators via the Lookout Support Knowledge portal.	

773

774 ### A.2.8 Microsoft Endpoint Configuration Manager (ECM) Support of Security
775 Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	Access to ECM Site Collections can be restricted via strong authentication. This can include MFA and passwordless options like Windows Hello for Business.	Ref1
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	ECM natively audits logins and activities and can be reported on by utilizing ECM Reports.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	ECM supports achieving least privilege through security roles, scopes, and collections.	Ref1
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	Microsoft provides guidance around the ports and protocols required by ECM. Customers can use this to implement firewalls between services and clients.	Ref1
SM 2.1	Does the software list and maintain an inventory of all software criticalities and integrations?	Configuration Manager uses an in-console service method called Updates and Servicing. It makes it easy to find and install recommended updates for your Configuration Manager infrastructure.	Ref1
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	ECM supports achieving least privilege through security roles, scopes, and collections.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	ECM supports encryption at rest natively and through the use of BitLocker.	Ref1
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	ECM supports encryption for data in transport.	Ref1
SM 2.5	Does the system support performing regular backups and restorations?	Backup and restore operations are core resiliency capabilities in ECM.	Ref1
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	ECM lists the software dependencies that are required for the platform to operate on the server in addition to client end nodes.	Ref1
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	Configuration Manager uses an in-console service method called Updates and Servicing. It makes it easy to find and install recommended updates for your Configuration Manager infrastructure.	Ref1
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	Configuration Manager supports installing specific roles, for example management points, distribution points, and software update points, which contain the services required to run that service only.	Ref1
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	Logs are stored in the ECM database, log files, and Windows Event Logs. Implementation guidance is specific to the capabilities of the SIEM.	
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	Configuration Manager includes software update monitoring, which can be used to identify vulnerable software on its infrastructure.	Ref1
SM 4.3	Is anti-malware or antivirus able to be installed on the system running your platform?	Anti-malware and anti-virus solutions can be installed on the host operating system. Microsoft recommends allowlisting the files and processes related to ECM.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 4.4	Does the platform allow for logging connection events to the tool?	Client and management point logging can be configured at various levels to meet customer requirements.	Ref1
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Microsoft offers training courses that are catered to the role an individual will have utilizing the product.	Ref1
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Microsoft provides e-learning and certification preparation guides for ECM on the Microsoft Learn portal. Hands-on or train-the-trainer models are provided through an implementation partner.	Ref1
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Courses and certifications are periodically updated based on product enhancements and feedback from customers.	

776

777 A.2.9 Tenable.sc Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.1	Does the software platform allow for the use of a two-factor authentication method for access?	MFA is achieved through certificate-based authentication and SAML authentication.	Ref1 Ref2
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	This is default behavior. Connections are authenticated with API keys or credentials, then handled via session cookie.	Ref1 Ref2
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	This is default behavior provided by role-based access control.	Ref1
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	Tenable.sc can bind https interface to a single IP/network interface card (NIC) and utilize sideband networks for management/administration.	Ref1 Ref2
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	This is default behavior provided by role-based access control.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	Tenable.sc provides encryption for critical resources (target credentials). For vulnerability data and application configuration information, an external data-at-rest solution is required.	Ref1
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	This is default behavior.	Ref1
SM 2.5	Does the system support performing regular backups and restorations?	Tenable supports administrator backup of the opt/sc directory. Backups can be scripted to run on the host OS.	Ref1
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	The Tenable.sc application can use the host OS's syslog implementation to leverage an external syslog or SIEM.	Ref1
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	Tenable.sc can scan an environment passively (with the use of Nessus Network Monitor/NNM) and actively to achieve continuous monitoring.	Ref1 Ref2
SM 4.3	Is anti-malware or antivirus able to be installed on the system running your platform?	Anti-malware and anti-virus solutions can be installed. Tenable recommends allowlisting the files and processes related to Nessus and Tenable.sc.	Ref1 Ref2
SM 4.4	Does the platform allow for logging connection events to the tool?	NNM not only does passive analysis for vulnerabilities, but it can also provide logging of connection events as Informational events.	Ref1
SM 4.5	Are there training courses or procedures in the event of an incident involving the tool or platform?	Tenable has many training options available to customers of our products, including instructional videos, free trainings, and paid trainings for deeper dives and larger groups.	Ref1 Ref2 Ref3
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	Tenable offers training courses that are catered to the role an individual will have utilizing the product.	Ref1 Ref2 Ref3

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	Tenable offers training courses for administrative users.	Ref1 Ref2 Ref3
SM 5.3	Are trainings updated and metrics collected to improve trainings?	Tenable continually collects feedback and introduces changes based on product updates and user feedback.	

A.2.10 VMware vRealize Automation SaltStack Config Support of Security Measures

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 1.2	Does the software platform identify and authenticate users and machine accounts that try to access the platform?	This can be set up in the SaltStack Config component or done through integration with LDAP, AD, SAML, or OpenID Connect (OIDC) providers.	Ref1
SM 1.3	Does the system allow for creating accounts with different access levels to enforce least management?	This can be set up in SaltStack Config or done through integration with LDAP, AD, SAML, or OIDC providers.	Ref1
SM 1.4	Does the system allow for the use of DACLs, network segmentation, or isolation for access to the platform?	The Linux OS hosting SaltStack Config can be configured to perform network isolation.	
SM 2.1	Does the software list and maintain an inventory of all software criticalities and integrations?	VMWare tracks each product used by SaltStack Config and any updates and vulnerabilities in those products.	
SM 2.2	Does the system allow for creating accounts with different access levels to enforce least management?	This can be set up in SaltStack Config or done through integration with LDAP, AD, SAML, or OIDC providers.	Ref1
SM 2.3	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data at rest?	SaltStack Config has a FIPS-compliant mode that can be configured at installation time to support encryption of data at rest.	Ref1
SM 2.4	Does the system use or contain an option to enable the use of NIST-certified cryptographic standards for protecting data in transit?	SaltStack Config supports encryption for data in transit by default. Key generation uses standard algorithms found in the OpenSSL library. These algorithms rely on OS-generated random seed data.	

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 2.5	Does the system support performing regular backups and restorations?	SaltStack Config allows administrators to perform manual backups.	Ref1
SM 3.1	Does the product list all software dependencies and currently installed applications/services?	SaltStack provides a list of all dependent software and libraries used within the product.	
SM 3.2	Does the platform allow for the deployment of patches and OS updates?	The Linux system hosting SaltStack can be updated by administrators. The SaltStack SecOps component can be utilized to perform updates on SaltStack nodes and client end nodes.	Ref1
SM 3.3	Does the platform allow for configuration management practices such as removal or disabling of services to maintain security?	SaltStack Config allows for configuration management through the implementation of Salt states, the beacon and reactor system, and/or orchestration.	Ref1
SM 4.1	Does the security tool support logging and sending that data to rsyslog or a SIEM?	Salt returners can be used/configured to send logs to third-party tools like rsyslog, splunk, etc.	Ref1
SM 4.2	Does the platform monitor the security and vulnerabilities associated with all software and dependencies used?	VMWare tracks each product used by SaltStack Config and tracks any updates and vulnerabilities that are announced by the product owners.	
SM 4.3	Is anti-malware or antivirus able to be installed on the system running your platform?	Anti-malware and anti-virus solutions can be installed on the host Linux OS.	
SM 4.4	Does the platform allow for logging connection events to the tool?	You can set the logging level to debug or turn on the audit trail, and that will provide connection events.	Ref1
SM 4.5	Are there training courses or procedures in the event of an incident involving the tool or platform?	There is official training for customers of the platform. Also, support contracts can be purchased to help troubleshoot and fix incidents with the product.	Ref1
SM 5.1	Are there training courses in how to use the products? Are there different courses for different roles?	VMware provides training on the underlying platform (SaltStack Config and vRealize Automation) as well as the security operations product.	Ref1

SM #	Question	Technical Mechanism or Configuration	Refs.
SM 5.2	Are there training courses for teaching administrators how to utilize the platform?	VMware provides training on the underlying platform (SaltStack Config and vRealize Automation) as well as the security operations product.	Ref1

Appendix B List of Acronyms

AD	Active Directory
AES	Advanced Encryption Standard
ANC	Adaptive Network Control
API	Application Programming Interface
BIOS	Basic Input/Output System
CAC	Common Access Card
CIO	Chief Information Officer
CISO	Chief Information Security Officer
CLI	Command-Line Interface
CRADA	Cooperative Research and Development Agreement
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
DACL	Discretionary Access Control List
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
ECM	(Microsoft) Endpoint Configuration Manager
EMM	Enterprise Mobility Management
EO	Executive Order
FAQ	Frequently Asked Questions
FedRAMP	Federal Risk and Authorization Management Program
FIPS	Federal Information Processing Standards
FMC	(Cisco) Firepower Management Center
FTD	(Cisco) Firepower Threat Defense
HA	High Availability

HSPD-12	Homeland Security Presidential Directive 12
IAM	Identity and Access Management
ICS	Industrial Control System
IoT	Internet of Things
IP	Internet Protocol
ISE	(Cisco) Identity Services Engine
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MDM	Mobile Device Manager
MES	(Lookout) Mobile Endpoint Security
MFA	Multi-Factor Authentication
NCCoE	National Cybersecurity Center of Excellence
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NNM	(Tenable) Nessus Network Monitor
OIDC	OpenID Connect
OS	Operating System
OT	Operational Technology
PC	Personal Computer
PIV	Personal Identity Verification
REST	Representational State Transfer
RMF	Risk Management Framework
SaaS	Software as a Service
SAML	Security Assertion Markup Language
SAN	Storage Area Network
SCCM	(Microsoft) System Center Configuration Manager

SGT	Security Group Tag
SIEM	Security Information and Event Management
SM	Security Measure
SMS	(Microsoft) Systems Management Server
SP	Special Publication
SPT	(Forescout) Security Policy Templates
SSH	Secure Shell
TLS	Transport Layer Security
UEM	Unified Endpoint Management
UI	User Interface
VLAN	Virtual Local Area Network
VM	Virtual Machine
WaaS	Windows as a Service
WSUS	(Microsoft) Windows Server Update Services

NIST SPECIAL PUBLICATION 1800-31C

Improving Enterprise Patching for General IT Systems:

Utilizing Existing Tools and Performing Processes in Better Ways

Volume C:
How-To Guides

Tyler Diamond*
Alper Kerman
Murugiah Souppaya
National Cybersecurity Center of Excellence
Information Technology Laboratory

Brian Johnson
Chris Peloquin
Vanessa Ruffin
The MITRE Corporation
McLean, Virginia

Karen Scarfone
Scarfone Cybersecurity
Clifton, Virginia

**Former employee; all work for this publication was done while at employer*

November 2021

DRAFT

This publication is available free of charge from
https://www.nccoe.nist.gov/projects/building-blocks/patching-enterprise

DISCLAIMER

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

While NIST and the NCCoE address goals of improving management of cybersecurity and privacy risk through outreach and application of standards and best practices, it is the stakeholder's responsibility to fully perform a risk assessment to include the current threat, vulnerabilities, likelihood of a compromise, and the impact should the threat be realized before adopting cybersecurity measures such as this recommendation.

National Institute of Standards and Technology Special Publication 1800-31C, Natl. Inst. Stand. Technol. Spec. Publ. 1800-31C, 123 pages, (November 2021), CODEN: NSPUE2

FEEDBACK

You can improve this guide by contributing feedback. As you review and adopt this solution for your own organization, we ask you and your colleagues to share your experience and advice with us.

Comments on this publication may be submitted to: cyberhygiene@nist.gov.

Public comment period: November 17, 2021 through January 10, 2022

All comments are subject to release under the Freedom of Information Act.

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Despite widespread recognition that patching is effective and attackers regularly exploit unpatched software, many organizations do not adequately patch. There are myriad reasons why, not the least of which are that it's resource-intensive and that the act of patching can reduce system and service availability. Also, many organizations struggle to prioritize patches, test patches before deployment, and adhere to policies for how quickly patches are applied in different situations. To address these challenges, the NCCoE is collaborating with cybersecurity technology providers to develop an example solution that addresses these challenges. This NIST Cybersecurity Practice Guide explains how tools can be used to implement the patching and inventory capabilities organizations need to handle both routine

and emergency patching situations, as well as implement workarounds, isolation methods, or other alternatives to patching. It also explains recommended security practices for patch management systems themselves.

KEYWORDS

cyber hygiene; enterprise patch management; firmware; patch; patch management; software; update; upgrade; vulnerability management

ACKNOWLEDGMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Peter Romness	Cisco
Matthew Hyatt	Cisco
John Loucaides	Eclypsium
Travis Raines	Eclypsium
Timothy Jones	Forescout
Tom May	Forescout
Michael Correa	Forescout
Jeffrey Ward	IBM MaaS360 with Watson
Joseph Linehan	IBM MaaS360 with Watson
Cesare Coscia	IBM MaaS360 with Watson
Jim Doran	IBM Research Team
Shripad Nadgowda	IBM Research Team
Victoria Mosby	Lookout

Name	Organization
Tim LeMaster	Lookout
Dan Menicucci	Microsoft
Steve Rachui	Microsoft
Parisa Grayeli	The MITRE Corporation
Yemi Fashina	The MITRE Corporation
Nedu Irrechukwu	The MITRE Corporation
Joshua Klosterman	The MITRE Corporation
Allen Tan	The MITRE Corporation
Josh Moll	Tenable
Chris Jensen	Tenable
Jeremiah Stallcup	Tenable
John Carty	VMware
Kevin Hansen	VMware
Rob Robertson	VMware
Rob Hilberding	VMware
Brian Williams	VMware

70 The Technology Partners/Collaborators who participated in this build submitted their capabilities in
71 response to a notice in the Federal Register. Respondents with relevant capabilities or product
72 components were invited to sign a Cooperative Research and Development Agreement (CRADA) with
73 NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Build Involvement
Cisco	Cisco Firepower Threat Defense (FTD) Cisco Identity Services Engine (ISE)
Eclypsium	Eclypsium Administration and Analytics Service
Forescout	Forescout Platform
IBM	IBM Code Risk Analyzer IBM MaaS360 with Watson
Lookout	Lookout Mobile Endpoint Security
Microsoft	Microsoft Endpoint Configuration Manager
Tenable	Nessus Tenable.io Tenable.sc
VMware	VMware vRealize Automation SaltStack Config

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly to conform to the publication and from which no deviation is permitted. The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited. The terms “may” and “need not” indicate a course of action permissible within the limits of the publication. The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory (ITL) draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication

or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

1. under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or
2. without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to: cyberhygiene@nist.gov

Contents

List of Figures

List of Tables

1 Introduction

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) built an example solution in a laboratory environment to demonstrate how organizations can use technologies to improve enterprise patch management for their general information technology (IT) assets.

This volume of the practice guide shows IT professionals and security engineers how we have implemented the example solution. It covers all of the products employed in this reference design, summarizes their integration into the laboratory environment, and documents security decisions and associated configurations. We do not re-create the product manufacturers' documentation, which is presumed to be widely available. Rather, these volumes show how we incorporated the products together in our environment.

This draft covers both phases of the example solution. Phase 1 involved two types of IT assets: desktop and laptop computers, and on-premises servers. Phase 2 added mobile devices and containers.

Note: These are not comprehensive tutorials. There are many possible service and security configurations for these products that are out of scope for this example implementation.

1.1 How to Use this Guide

This NIST Cybersecurity Practice Guide demonstrates a standards-based example solution and provides users with the information they need to replicate the proposed approach for improving enterprise patching practices for general IT systems. This design is modular and can be deployed in whole or in part.

This guide contains three volumes:

- NIST Special Publication (SP) 1800-31A: *Executive Summary* – why we wrote this guide, the challenge we address, why it could be important to your organization, and our approach to solving the challenge
- NIST SP 1800-31B: *Security Risks and Capabilities* – why we built the example implementation, including the risk analysis performed and the security capabilities provided by the implementation
- NIST SP 1800-31C: *How-To Guides* – what we built, with instructions for building the example implementation, including all the details that would allow you to replicate all or parts of this project **(you are here)**

Depending on your role in your organization, you might use this guide in different ways:

Business decision makers, including chief security and technology officers, will be interested in the *Executive Summary, NIST SP 1800-31A*, which describes the following topics:

- challenges that enterprises face in mitigating risk from software vulnerabilities
- example solution built at the NCCoE
- benefits of adopting the example solution

Business decision makers can also use *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology*. It complements the implementation focus of this guide by recommending creation of an enterprise strategy to simplify and operationalize patching while also reducing risk.

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in *NIST SP 1800-31B*, which describes what we did and why. The following sections will be of particular interest:

- Section 3.5.1, Threats, Vulnerabilities, and Risks, describes the risk analysis we performed.
- Section 3.5.2, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices.

You might share the *Executive Summary, NIST SP 1800-31A*, with your leadership team members to help them understand the importance of adopting standards-based, automated patch management. Also, *NIST SP 800-40 Revision 4 (Draft), Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology* may also be helpful to you and your leadership team.

IT professionals who may be interested in implementing an approach similar to ours will find the entire practice guide useful. In particular, the How-To portion of the guide, *NIST SP 1800-31C* could be used to replicate all or parts of the build created in our lab. Furthermore, the How-To portion of the guide provides specific product installation, configuration, and integration instructions for implementing the example solution. We have omitted the general installation and configuration steps outlined in manufacturers' product documentation since they are typically made available by manufacturers. Instead, we focused on describing how we incorporated the products together in our environment to create the example solution.

This guide assumes that the reader of this document is a seasoned IT professional with experience in implementing security solutions within an enterprise setting. While we have used a suite of commercial and open-source products to address this challenge, this guide does not endorse these particular products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of an automated enterprise patch management system. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and recommended practices. The Technologies section of *NIST SP 1800-31B* lists the products we used and maps them to the cybersecurity controls provided by this example solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution, but an example solution. This is a draft guide. We seek feedback on the contents of this guide and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to cyberhygiene@nist.gov.

1.2 Build Overview

This NIST Cybersecurity Practice Guide addresses the use of commercially available technologies to develop an example implementation for deploying an automated patch management system. This project focuses on enterprise patch management for small to large enterprises. The example solution demonstrates how to manage assets to reduce outages, improve security, and continuously monitor and assess asset vulnerabilities.

1.2.1 Use Case Scenarios

The NCCoE team worked with the project collaborators to create a lab environment that includes the architectural components and functionality that will be described later in this section. These use case scenarios were demonstrated in the lab environment as applicable for desktop and laptop computers, on-premises servers, mobile devices, and containers:

- **Asset identification and assessment**: discovering physical and virtual assets on your corporate network, and performing automated assessments to prioritize their remediation. For this scenario, it is important to determine some information about each asset, such as hostname, Internet Protocol (IP) address, Media Access Control (MAC) address, firmware version, operating system (OS) version, and installed software packages. This information can be used to identify the asset and synchronize with other systems such as asset and configuration management tools. Once the asset has been identified, it is important to determine if the software and firmware versions have known vulnerabilities and how critical those vulnerabilities are. The collected information is categorized and integrated with other asset and configuration management tools.
- **Routine patching**: modifying assets to configure and install firmware, OSs, and applications for the purpose of addressing bug fixes, providing security updates, and upgrading to later, supported releases of software. Routine patching is done on regularly scheduled intervals defined by the organization.
- **Emergency patching**: performing emergency patching of assets, such as for an extreme severity vulnerability or a vulnerability being actively exploited in the wild. Systems in this scenario should be able to deploy patches to assets outside of regularly scheduled intervals.
- **Emergency workaround**: implementing emergency workarounds for identified assets, such as temporarily disabling vulnerable functionality. This scenario demonstrates an emergency procedure in which an organization needs to temporarily mitigate a vulnerability prior to a

vendor releasing a patch. Systems included in this scenario need to be able to uninstall, reconfigure, and disable services on assets.

- **Isolation of unpatched assets**: performing network isolation of assets, like unsupported legacy assets, end-of-life assets, and assets with high operational uptime requirements, to mitigate risk for assets that cannot be easily patched or cannot be patched at all.
- **Patch management system security**: implementing recommended security practices for patch management systems, which have administrative privileged access over many other systems. See Section 3 of *NIST SP 1800-31B* for more information on addressing this scenario.

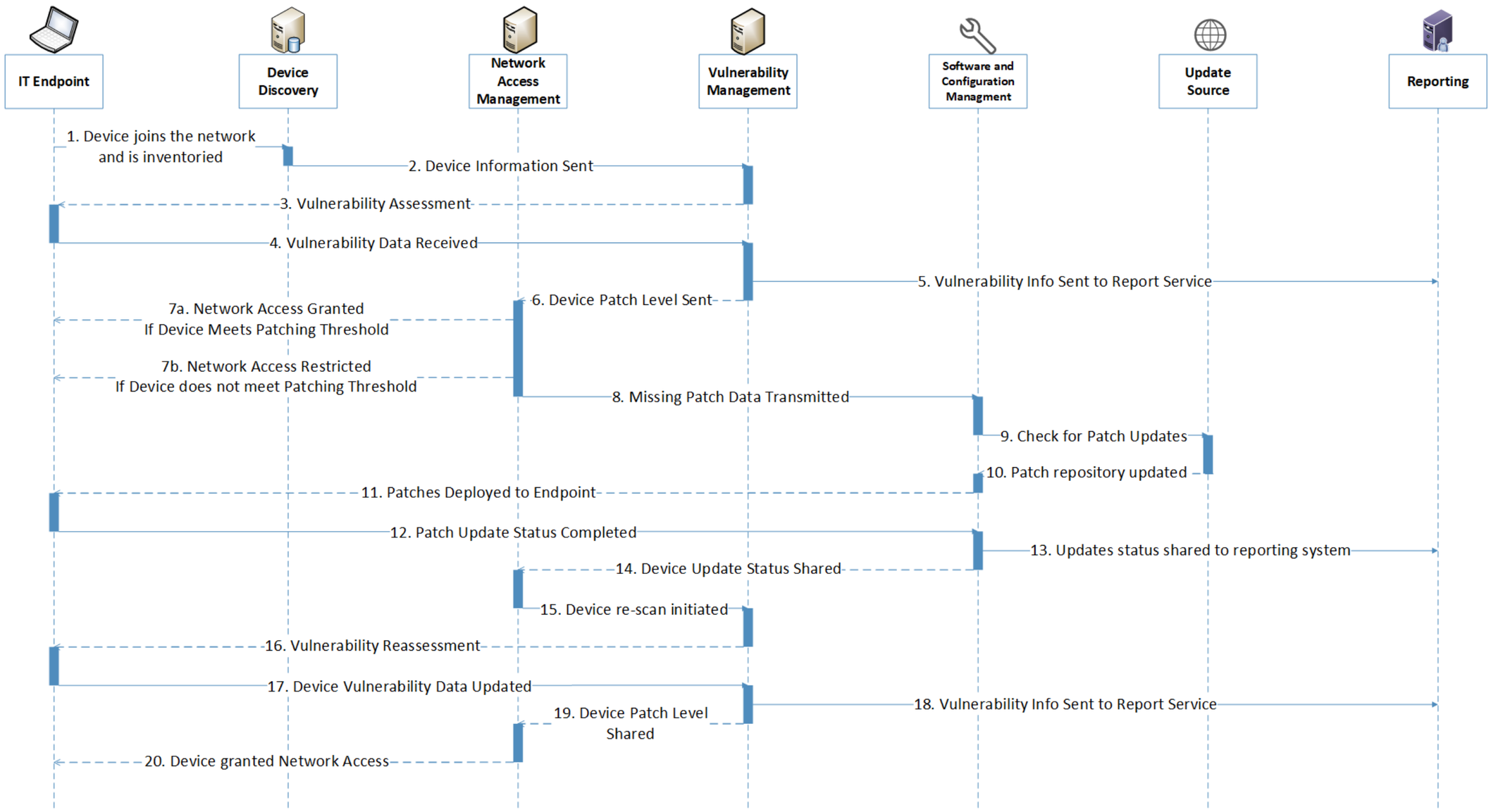
1.2.2 Logical Architecture

This project required a variety of technology capabilities. The following were included in the lab build:

- **IT endpoints**: This represents traditional endpoints, which included Apple laptops, Linux workstations/servers, and Windows workstations/servers, as well as newer types of endpoints, such as containers and Android and iOS mobile devices. These endpoints were all integrated either physically or virtually within the network environment.
- **Device discovery**: This includes systems that actively or passively scan the network environment and report about newly discovered assets and their observed characteristics.
- **Network access management**: This includes systems that govern access for endpoints, which are components that typically enforce access restrictions based on telemetry received from device discovery and vulnerability management systems within the environment. For example, enterprise assets that are not up to the required patch levels could be restricted from having access to resources distributed across the network environment.
- **Vulnerability management**: This includes systems that continually scan endpoints to identify known vulnerabilities and associated risks so that they may be proactively mitigated through appropriate patching and configuration settings.
- **Software and configuration management:** This includes systems that automate and maintain configuration changes and consistency across endpoints within the environment, as well as update currently installed software and firmware versions. Configuration changes may include updating network information, installing/uninstalling programs and services, and starting and stopping services.
- **Update sources**: This includes systems that house and maintain the most recent and trusted software updates/upgrade files for distribution within the environment. These update sources were leveraged by the software distribution systems to maintain an updated repository of available patches.
- **Reporting**: This includes systems that collect information from device discovery, network access management, and vulnerability management systems. This collected information can then be presented via dashboards or reports.

Figure 1-1 depicts the components that are used in the logical architecture, and the flow of a new or returning device joining the network.

405 **Figure 1-1 Logical Architecture Components and Flow**



The following steps take place as a new or returning device joins the network. Each number corresponds to a flow in Figure 1-1.

Device discovery: 1) The device discovery tool scans the device and collects information such as IP address, MAC address, installed software/firmware, and OS, then 2) sends the information to a vulnerability management system.

Vulnerability scanning: 3) The vulnerability management system scans the endpoint for vulnerability information, including missing patches and outdated software, and 4) receives the scan results. 5) The vulnerability management system sends the collected vulnerability data to the reporting service for presentation to administrators.

Quarantine decision and enforcement: 6) The vulnerability management system shares the device patch level with the network access management system to be used for network access control. 7) The network access management system applies one of the following two enforcement actions: 7a) If the network device does not exceed the organizational patch threshold, the device is given network access and does not need to go through the remainder of the diagram. 7b) If the network device exceeds the organizational patch threshold, the network access management system performs quarantine actions on the endpoint and restricts network access. 8) The network access management system shares the missing patch information with the software and configuration management system.

Patching: 9) The software and configuration management system checks its trusted update source for patch updates, then 10) receives any new patches and updates its patch repository database. 11) Missing patches are deployed from the software and configuration management system to the connected endpoint. 12) The software and configuration management system receives the update that the patches have been installed successfully. 13) The updates that were applied are sent to a reporting server for administrator review. 14) The software and configuration management system communicates that updates were successfully applied to the endpoint.

Vulnerability scanning: 15) The network access management system initiates a rescan of the endpoint by communicating with the vulnerability management system. 16) The vulnerability management system rescans the endpoint and 17) collects updated vulnerability data. 18) The vulnerability management system sends updated endpoint vulnerability data to the reporting server and 19) shares device patch level information with the network access management server.

Network access granted: 20) The network access management server grants the endpoint network access.

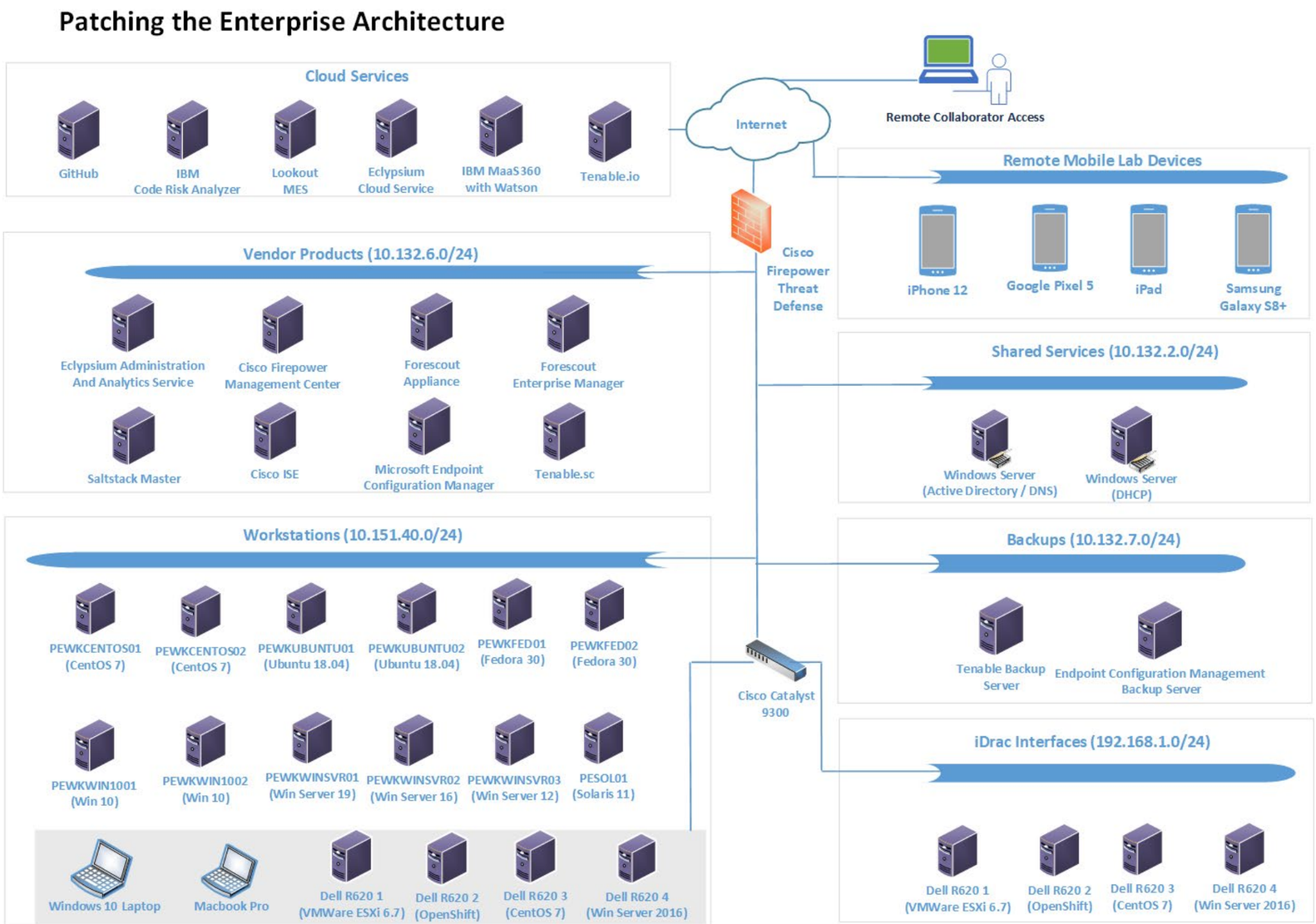
1.3 Build Architecture Summary

Figure 1-2 depicts the high-level physical architecture of the NCCoE laboratory environment. The segmented laboratory network backbone models the separation that typically exists between subnetworks belonging to different parts of an enterprise, such as a backup site, shared services, a data

center hosting widely used applications and services, and a workstation subnet consisting of user endpoints. While the majority of the nodes in the workstation subnet were virtual, the gray box notes physical machines.

The subnets were extended from the virtual lab to the physical lab by a Cisco switch. The switch had the workstation virtual local area network (VLAN) extended to it from VMWare via a trunk port. The lab subnetworks were connected by a Cisco Firepower Threat Defense (FTD) firewall.

447 **Figure 1-2 Laboratory Configuration of Example Solution Architecture**



The NCCoE lab provided the following supporting infrastructure for the example implementation:

- VMWare ESX version 7.0, a shared NCCoE resource provided by the NCCoE IT Operations team to host the patching infrastructure's virtual machine (VM) workloads and network infrastructure
- a dedicated VLAN provided for external collaborator remote access to the VMWare lab environment from NCCoE IT operations
- a Windows 2016 server that provided Active Directory (AD) services, authenticated users and machines to the lab.nccoe.org domain, and provided Domain Name System (DNS) services
- a Windows 2019 server that provided Dynamic Host Configuration Protocol (DHCP) services to the endpoint network
- a Windows 2019 server that served as a remote backup site for the endpoint configuration management system
- a CentOS 7 machine that served as a remote backup site for the Tenable vulnerability management system
- iDrac interfaces that allowed for remote configuration of Dell R620 server blades
- virtualized endpoints running the following OSs: CentOS, Fedora, macOS, Redhat, Solaris, Ubuntu, Windows Enterprise, and Windows Server
- a physical Windows 10 laptop and a physical Apple laptop running macOS to represent employee endpoints
- a Microsoft SQL server hosting the database for Microsoft Endpoint Configuration Manager
- several Dell R620 machines that had Windows Server 2016, VMWare ESXi, and two machines running CentOS 7 installed to represent physical end nodes. Of the two CentOS 7 machines, one was chosen to have OpenShift installed to represent a container management platform. The Docker repository was also run on this same OpenShift machine.

1.4 Implemented Products and Services

The following collaborator-supplied components were integrated with the supporting infrastructure to yield the example implementation:

- **Cisco Firepower Management Center (FMC)** version 6.5.0.4 provided centralized management of the Cisco Firepower Threat Defense firewall. It supplied a web interface for firewall administrators.
- **Cisco Firepower Threat Defense (FTD)** version 6.4.0 was the central firewall that connected the lab's internal subnets and the external internet. Through communication with Cisco Identity Services Engine (ISE), the firewall provided network segmentation capabilities.

- **Cisco Identity Services Engine (ISE)** version 2.7.0.36 was utilized to perform asset inventory and discovery. Using attributes that were collected by Cisco ISE, such as current user or patch level, the firewall enforced custom network access control policies.
- **Eclypsium Administration and Analytics Service** version 2.2.2 was configured to assess firmware levels present on a device and report if a vulnerable version of firmware was running on a device. It could then download firmware updates to affected devices.
- **Forescout Platform** version 8.2.2 provided asset inventory and discovery. Additionally, Forescout collected attributes associated with endpoints and, through policy, provided enforcement actions such as network access control via an integration with pxGrid, or service removal via custom scripts.
- **IBM Code Risk Analyzer** (cloud-based service) provided vulnerability scanning and reporting for source code as part of the DevOps pipeline. Through an integration with GitHub, it scanned deployed code for vulnerabilities and produced a report of remediation actions. IBM, as part of the lab effort, provided source code hosted in GitHub to be ingested by the Code Risk Analyzer.
- **IBM MaaS360 with Watson** (cloud-based service) provided asset inventory, vulnerability management, and software distribution to laptops and mobile devices. The user authentication module, part of the Cloud Extender module, was used to integrate IBM MaaS360 with AD. This allowed users to authenticate to MaaS360 with their domain-joined accounts.
- **Lookout Mobile Endpoint Security** (cloud-based service) provided vulnerability scanning, assessment, reporting, and policy enforcement for mobile devices. An integration with IBM MaaS360 allowed custom attributes from Lookout to be used in MaaS360 policies.
- **Microsoft Endpoint Configuration Manager** version 2002 provided device configuration and software distribution capabilities. Endpoint Configuration Manager allowed for software updates and software changes to be pushed to endpoints. Discovery capabilities were enabled to determine what endpoints existed on the network and domain.
- **Nessus** version 8.14.0 provided on-premises vulnerability scanning of the architecture. Nessus logged into devices over the network, using supplied credentials, and enumerated vulnerabilities and missing patch information. This information was then presented to the administrator via the managing Tenable.sc tool.
- **Tenable.io** (cloud-based service) provided vulnerability scanning and reporting for containerized applications. Tenable.io was configured to upload a repository from an OpenShift node and perform assessments.
- **Tenable.sc** version 5.18.0 provided management of the lab Nessus scanner. Tenable.sc was configured to utilize the Nessus scanner to provide on-premises vulnerability scanning, asset inventorying/discovery, and reporting using dashboards. Scan data from Tenable.sc was ingested by other systems and was exported in the form of reports.
- **VMware vRealize Automation SaltStack Config** version 8.3.0 provided device configuration and software distribution capabilities. SaltStack Config allowed for configuration changes to be

made to devices by updating or removing software as well as updating settings such as network information.

Table 1-1 lists the collaborator-supplied product versions and system configurations that were utilized in the implementation, including the number of central processing units (CPUs) and the amount of random access memory (RAM) and hard disk drive (HDD) space in gigabytes (GB). All products were either deployed virtually via an Open Virtualization Appliance (OVA) or installed on VMs. In addition to these products, five cloud-based software as a service (SaaS) offerings were also used for the build: IBM Code Risk Analyzer, IBM MaaS360 with Watson, Lookout Mobile Endpoint Security, Tenable.io, and a SaaS version of Eclypsium.

Table 1-1 Product Versions and System Configurations Used

Product	Version	OS	CPUs	RAM	HDD	Deployed Via
Cisco FMC	6.5.0.4	N/A	4	32 GB	250 GB	OVA
Cisco FTD	6.4.0	N/A	4	8 GB	49 GB	OVA
Cisco ISE	2.7.0.36	N/A	2	8 GB	200 GB	OVA
Eclypsium Administration and Analytics Service (on-premises)	2.2.2	CentOS 7	2	8 GB	200 GB	Installed application
Forescout Appliance	8.2.2	N/A	6	14 GB	200 GB	OVA
Forescout Enterprise Manager	8.2.2	N/A	4	12 GB	200 GB	OVA
Microsoft Endpoint Configuration Manager	2002	Windows Server 2019	4	8 GB	240 GB	Installed application
Nessus	8.14.0	CentOS 7	2	8 GB	200 GB	Installed application
Tenable.sc	5.18.0	CentOS 7	2	8 GB	80 GB	Installed application
VMware vRealize Automation SaltStack Config	8.3.0	CentOS 7	2	12 GB	80 GB	Installed application

Sections 2 through 9 of this volume contain more information on each of these products and services, grouped by vendor. Note that the vendor sections are in order by the approximate sequence followed in this build for installing and configuring the products and services.

1.5 Supporting Infrastructure and Shared Services

In the lab environment, common services were deployed to support the example solution. These services included AD Domain Services, Windows DNS, Windows DHCP, and a physical Cisco switch.

1.5.1 AD Domain Services

The AD Domain Services deployment provided the directory services that many of the products relied on for their installations. A directory stores information about objects such as users and computers. This information is made accessible on the network and can be used for many purposes; in this reference implementation it was mainly used for authentication and access control. The AD Domain Services instance in our reference implementation was deployed on a single VM running Windows Server 2016. This server was accessible to all subnets on the lab. More information about AD Domain Services and the capabilities it provides can be found here.

1.5.2 Windows DNS

The Windows DNS deployment provided DNS capabilities to the reference implementation. DNS is an open protocol that is primarily used to translate domain names to IP addresses. The Windows DNS instance in our reference implementation was deployed on the same Windows Server 2016 VM running AD Domain Services. This server was accessible to all subnets of the lab, giving all computers access to DNS. More information on how to deploy Windows DNS can be found here.

1.5.3 Windows DHCP

The Windows DHCP deployment provided DHCP capabilities to the endpoints located in the Workstation network segment. DHCP is a network management protocol that is primarily used to provide network parameters, such as an IP address and default gateway, to endpoints. The Windows DHCP instance in our reference implementation was deployed on a Windows 2019 server VM. This server was accessible to the endpoint subnet of the patching architecture, giving all computers connected to the endpoint subnet access to DHCP. More information on how to deploy Windows DHCP can be found here.

1.5.4 Cisco Switch

The architecture utilized a Cisco Catalyst 9300 switch to extend the VMWare VLANs to the physical devices within the lab environment, including laptops and server blades. A trunk port configured on the switch allowed for the VLANs configured in VMWare to be recognized by the switch. The remaining switch ports were configured to access one VLAN at a time, depending on the connected device. More information on the Cisco Catalyst 9300 switch can be found here.

1.6 Typographic Conventions

The following table presents typographic conventions used in this volume.

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For language use and style guidance, see the *NCCoE Style Guide*.
Bold	names of menus, options, command buttons, and fields	Choose **File** > **Edit.**
`Monospace`	command-line input, onscreen computer output, sample code examples, and status codes	`mkdir`
`Monospace Bold`	command-line user input contrasted with computer output	**`service sshd start`**
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

2 Tenable

In the first phase of our build we used Tenable products to provide on-premises vulnerability scanning, asset inventorying/discovery, and reporting using dashboards. Tenable was leveraged to meet the device discovery, software/firmware discovery, and software/firmware assessment scenarios. Two Tenable products, Nessus Scanner and Tenable.sc, were used in the lab environment as part of this project. Also, Tenable.io, a SaaS-based cloud offering from Tenable, provided vulnerability scanning of container images to the lab environment during the second phase of the build. This section shows how each product was installed, configured, and used in the lab.

2.1 Nessus Installation and Configuration

Nessus is a vulnerability scanning engine that is used to scan endpoints, such as Linux, Windows, and macOS, VMWare ESXi, and network switches for vulnerability data. We utilized Nessus to scan endpoints for vulnerability information and feed this information to Tenable.sc for reporting. Nessus can be deployed as a standalone server or managed by Tenable.sc. In our lab build, Nessus was managed by Tenable.sc. Since Nessus needed to be linked to Tenable.sc during Tenable.sc's setup, Nessus was installed and set up first.

Nessus was installed on a CentOS 7 VM, with hardware details included in Section 1.4. More information on Nessus requirements can be found here. Installing Nessus 8.14.0 consisted of the following steps (with more detailed information available from the hyperlinked resources):

1. Download the Nessus executable from the Tenable download page. Note that you will need a Tenable account to download installation software.
2. Install Nessus by running the rpm installation command, then start the Nessus service.
3. Configure Nessus to be managed by Tenable.sc after installing Tenable.sc.

2.2 Tenable.sc

Tenable.sc is a vulnerability management product that collects information from Nessus and reports that information to administrators using dashboards and reports. Our build utilized Tenable.sc to manage a Nessus scanner and report on collected vulnerability data for scanned endpoints. This section assumes that the Nessus scanner from Section 2.1 was installed before installing Tenable.sc.

2.2.1 Tenable.sc Installation and Configuration

Tenable.sc was installed on a CentOS 7 VM, with hardware details included in Section 1.4. The Tenable site has more information on Tenable.sc requirements. Installing and configuring Tenable.sc 5.18.0 consisted of the following steps:

1. Download Tenable.sc from the Tenable site (note: a Tenable account is needed).
2. Install Tenable.sc using the appropriate rpm command and start the Tenable.sc service.
3. License Tenable.sc.
4. Configure Tenable.sc:
    a. Add a Nessus scanner. Tenable.sc relies on vulnerability data collected from Nessus scanners to provide information on endpoint vulnerability levels.
    b. Add a repository. A repository holds vulnerability data that is collected from Nessus scanners for organizational endpoints. Repositories provide data storage that can be restricted to appropriate users.
    c. Add an organization. Organizations provide logical groupings for Tenable resources. Administrators can restrict access to organizations to ensure that only authorized personnel can view data.
    d. Add a user with Security Manager permissions. The Security Manager role needs to be added before a scan can be run. By default, when installing Tenable.sc a local system administrator account is created, and that account is responsible solely for setting up organizations and repositories. A Security Manager account has the correct permissions to view scan data and initiate scans. More information on other Tenable.sc security roles can be found here.

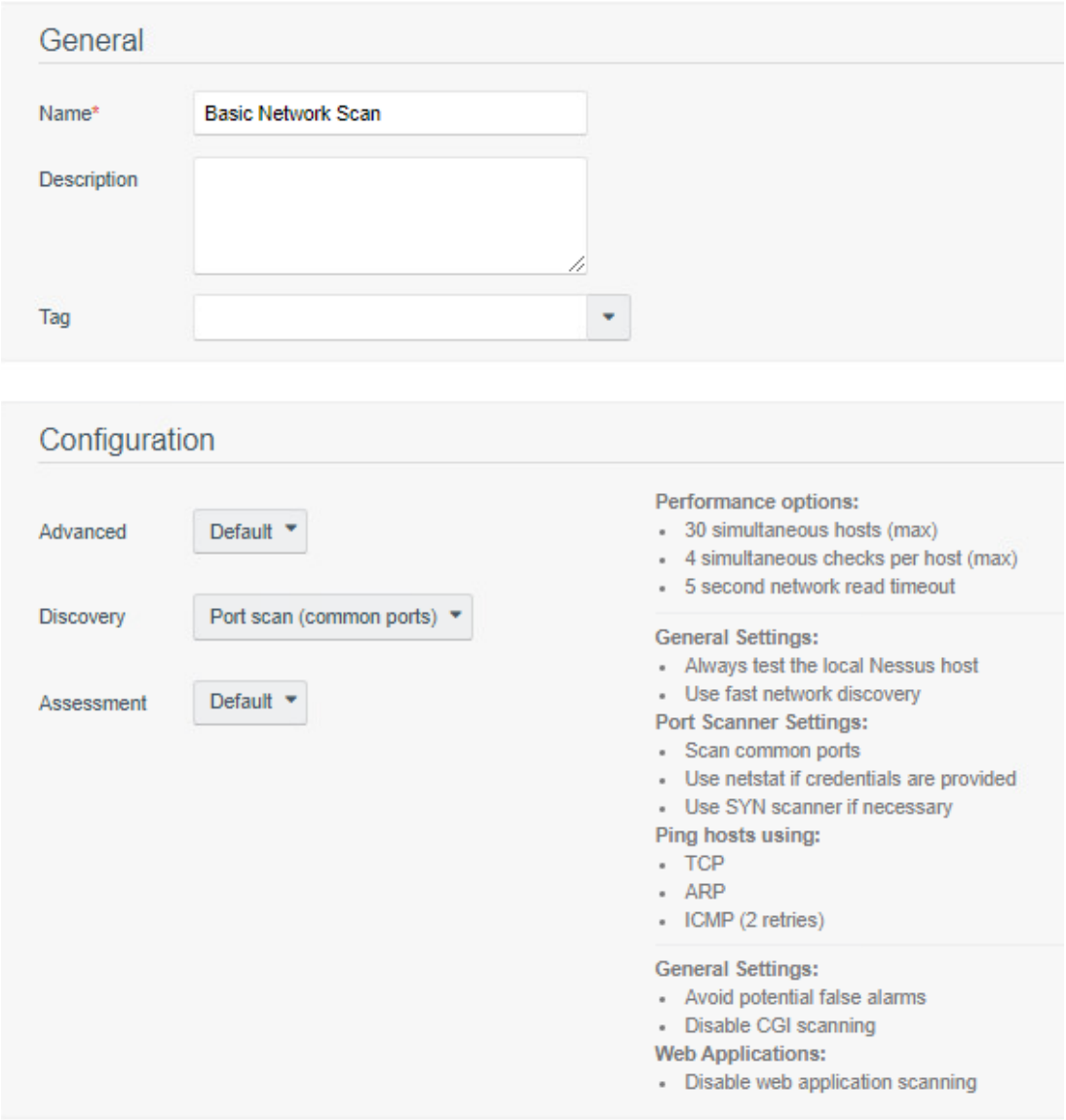
e. Add endpoint credentials. Tenable.sc requires credentials to be loaded in order to obtain the correct access levels for vulnerability scan data to be collected. Missing results may be observed by scanning an endpoint without credentials. More information on credentials can be found here.

2.2.2 Tenable.sc Scan Setup and Launch

After installing Nessus and Tenable.sc, the next step was to set up a scan policy. Scan policies allow you to deploy template-based or custom scan options for assessing endpoints, including Windows, VMWare ESXi, macOS, and Linux-based OSs, as well as networking equipment. Scan policies contain plugin settings and other advanced options that are used during active scans. For our build, Tenable recommended the Basic Network scan template with credentials to assess vulnerabilities, because it performs a full system scan that is suitable for a variety of hosts regardless of OS. Our build performed a credentialed scan to help Tenable enumerate missing patch information; other options were non-credentialed scans and agent-based scanning. More information on other types of Tenable.sc scan templates and when they may be used can be found here.

We used the options below when creating our scan policy. See https://docs.tenable.com/tenablesc/Content/AddScanPolicy.htm for more information on adding scan policies.

- **Template**: Basic Network scan
- **Name**: Lab Basic Scan
- **Advanced**: Default
- **Discovery**: Port scan (common ports)
- **Assessment**: Default
- The **Report** and **Authentication** tabs stayed at their default values, as credentials will be added in the active scan section.



637

638 The next step after creating a scan policy was to add that policy to an active scan. Active scans utilize the
639 scan policy as well as user-supplied options to launch scans against endpoints. More information on
640 creating an active scan is available here. We used the following options when creating our active scan:

641 - **Name**: Credentialed Scan
642 - **Policy**: Lab Basic Scan
643 - **Schedule**
644 - **Frequency**: Weekly

- **Time**: 03:00
- **Timezone**: America/New_York
- **Repeat Every**: Saturday

- **Import Repository**: Patching Lab Endpoints
- **Target Type**: IP/DNS Name
- **IPs / DNS Names**: 10.151.40.0/24
- **Credentials**: Add all credentials created in step

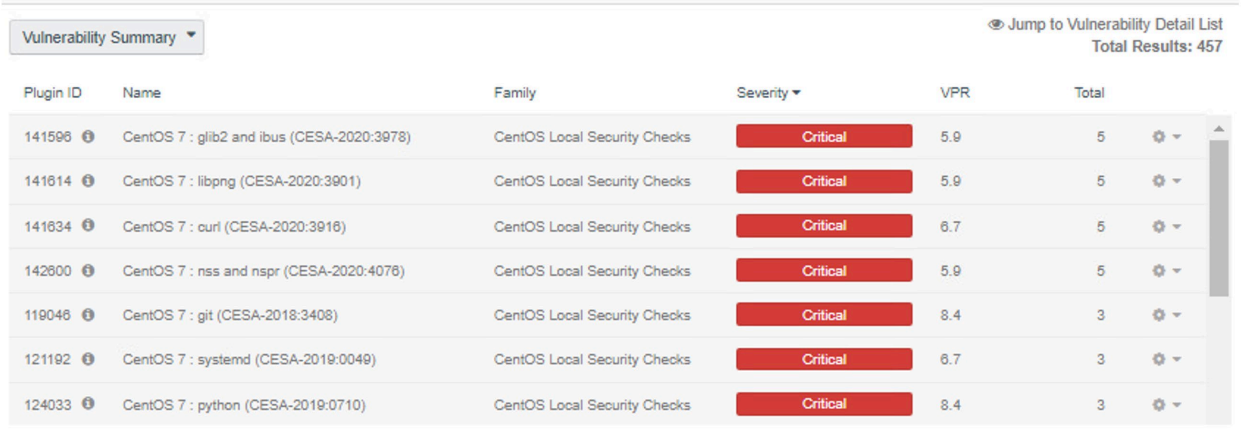
After creating the active scan, click **Submit**. The example above would be scheduled to run automatically on Saturdays at 3 a.m.

Information on manually launching scans (ad-hoc) is available here.

2.2.3 Scan Results

By default, when viewing scan results, the user is taken to the vulnerability summary page. This page contains information on observed vulnerabilities, and the results are sorted by observed Common Vulnerability Scoring System (CVSS) severity and the number of observed affected machines. Figure 2-1 shows vulnerability summary information from our build. The vulnerabilities can be viewed by package name and OS. The scan results can also be sorted by different types, such as IP address. This can be useful in allowing administrators to quickly see which vulnerabilities were discovered per asset.

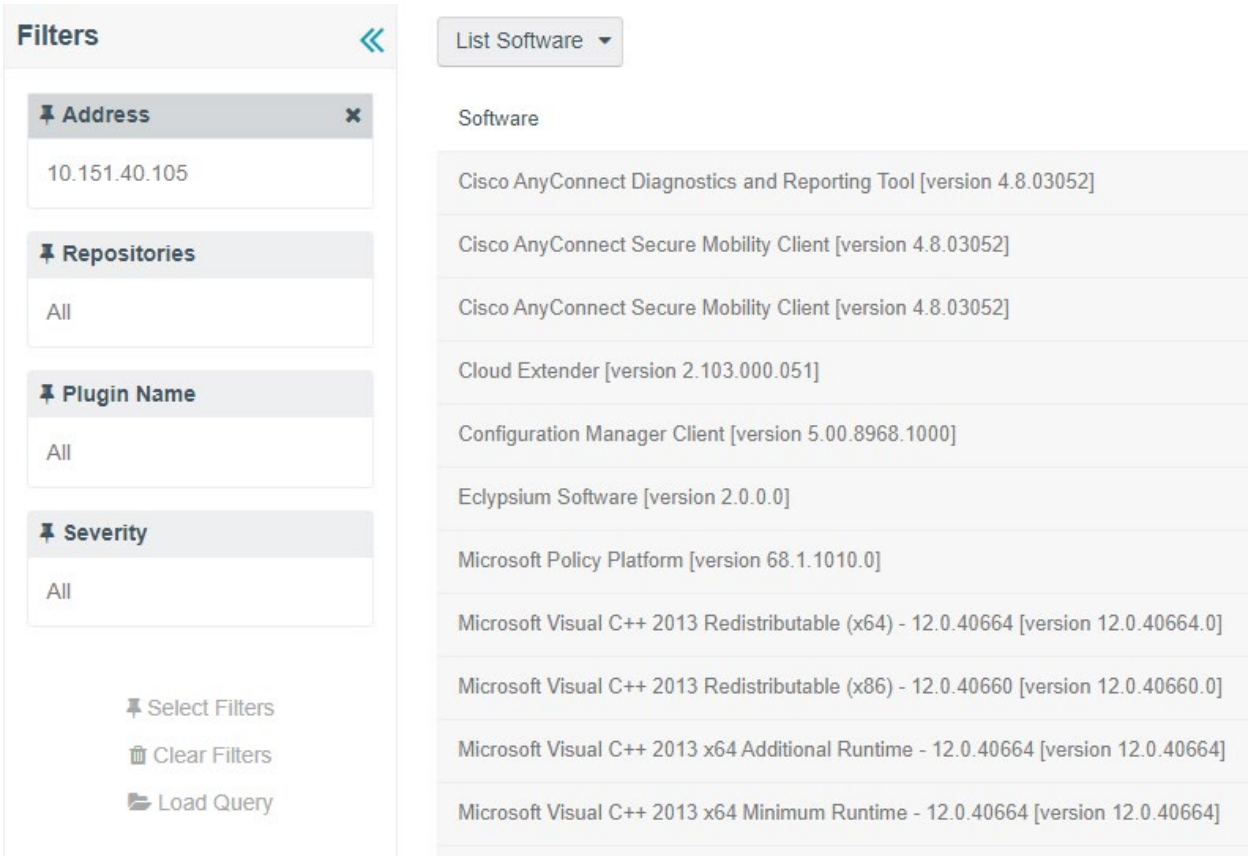
Figure 2-1 Vulnerability Summary Information



Sorting by IP Summary and then clicking the IP address of a machine allows for additional filters to be applied to scan results. Another filter that could be utilized for software discovery is clicking on **List**

665 **Software** while searching for a specific IP address. This filter shows all of the software that is currently
666 running and discovered on a machine, as the example in Figure 2-2 illustrates.

667 **Figure 2-2 Applying Filters to Scan Results**

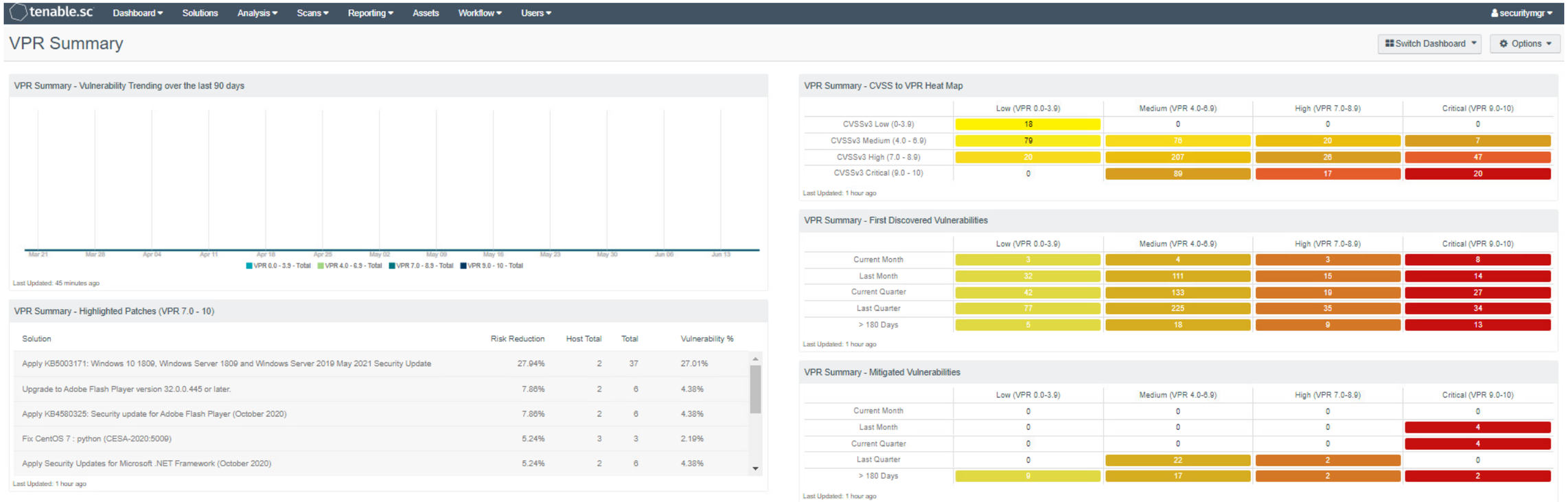


668 ### 2.2.4 Tenable.sc Dashboards

669 Tenable.sc provides graphical representations of information that is obtained via vulnerability scans.
670 Dashboards can be customized with different widgets to allow organizations to quickly observe
671 vulnerability information. We utilized Tenable.sc's reporting dashboards to help prioritize which assets
672 to remediate first and meet the firmware and software assessment scenarios. Directions for adding a
673 dashboard are available here. We used two dashboards: the Vulnerability Prioritization Rating (VPR)
674 Summary dashboard and the Worst of the Worst - Fix These First! dashboard.

675 The VPR Summary dashboard was utilized to help administrators prioritize which systems in the lab
676 should be remediated first. VPR combines threat intelligence, machine learning, research insights, and
677 vulnerability metrics to dynamically measure risk. A higher number on the VPR dashboard indicated
678 which systems should be immediately addressed. Figure 2-3 shows the VPR dashboard from the build.

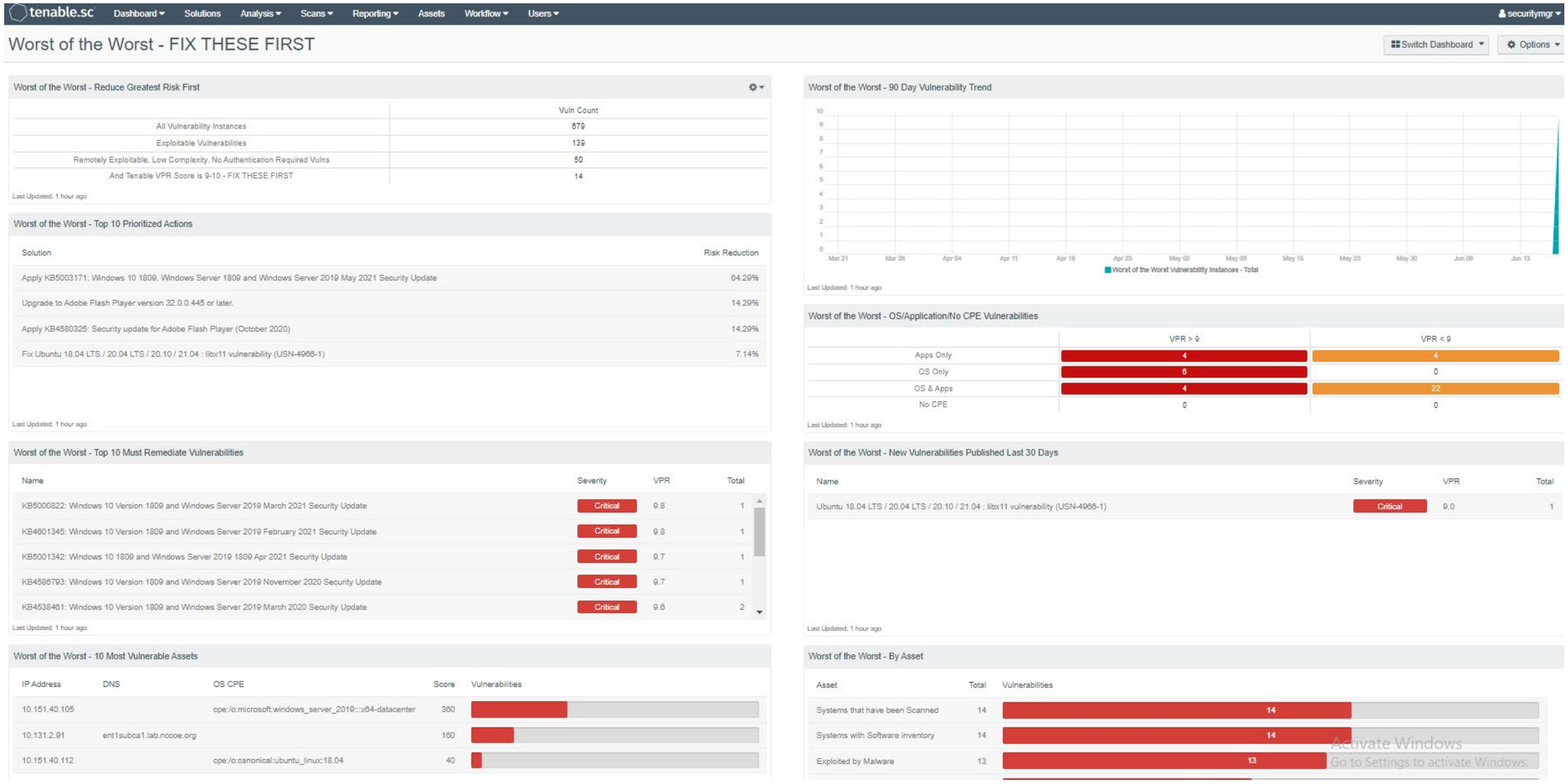
679 **Figure 2-3 Tenable VPR Summary Dashboard**



680

The Worst of the Worst – Fix These First! dashboard was used to help system administrators prioritize remediation efforts. The dashboard allows system administrators to gain insight into the top 10 vulnerabilities affecting systems and the top 10 remediation actions that should be taken. The dashboard also shows a list of the most vulnerable assets. Figure 2-4 shows an example of the Worst of the Worst dashboard, with the top 10 most vulnerable assets and exploitable vulnerabilities.

Figure 2-4 Tenable Worst of the Worst – Fix These First! Dashboard Example

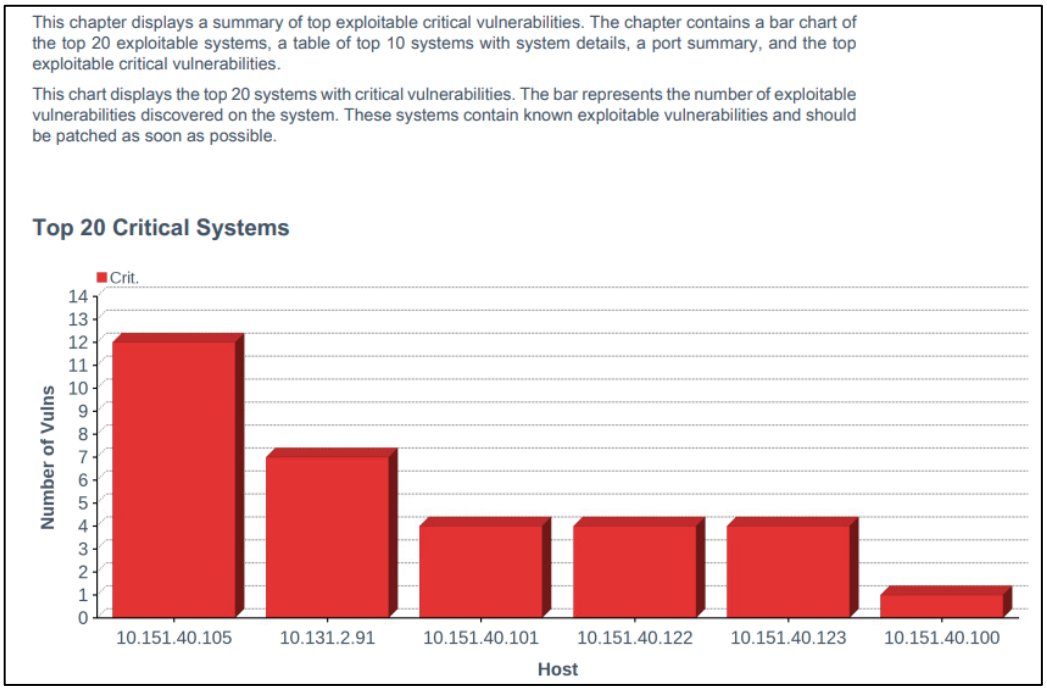


2.2.5 Tenable.sc Reporting

Tenable.sc also provides the ability to export vulnerability data to reports. The difference between dashboards and reports is that reports are meant to be exported and used outside of the Tenable.sc web console. With reports, data can be exported as a comma-separated values (CSV) file for ingestion by other systems, or as PDF files to be reviewed by management for compliance or vulnerability management purposes. Our build utilized reports to demonstrate how software and firmware assessment data could be shared with security managers to help to prioritize remediation efforts and actions.

Tenable reports can be scheduled to run after a scan or be scheduled to run during certain times of the week. To launch a report on demand (manually start), follow the instructions here. Once the report is ready and the results are clicked on, the report will automatically download in the browser. Figure 2-5 shows a portion of the Critical and Exploitable Vulnerabilities report from our build that detailed the top 20 critically affected systems.

Figure 2-5 Exploitable Vulnerability Summary



More information about reports, other report templates, and custom report creation can be found here.

2.2.6 Tenable.sc Integrations

Tenable.sc provides for integrations with third-party software via its representational state transfer (REST) application programming interface (API). The vulnerability data that is collected by Tenable can be shared with other systems such as configuration management or access control systems to automatically apply remediation actions. More information on the Tenable API can be found here. The following two example integrations with Tenable.sc were implemented in the lab:

- **Cisco ISE**: This integration allowed Cisco ISE to leverage vulnerability data collected by Tenable.sc. Cisco ISE would initiate a scan when new devices joined the network. The CVSS scores observed by Tenable were then sent to Cisco ISE, and devices that were over the score threshold were automatically quarantined from internal access. See Section 5.2.5 for additional information on the Cisco ISE integration.
- **Forescout Platform**: This integration allowed Forescout to leverage vulnerability data collected by Tenable.sc in order to quarantine endpoints. Forescout policy was created that specified that devices with CVSS scores over a certain threshold would be quarantined from the network. Forescout leveraged an integration with Cisco ISE via pxGrid to perform network enforcement actions. Section 7.2.8 contains additional explanation of the integration.

2.2.7 Tenable.sc Ongoing Maintenance

All Tenable components should be kept up to date. You must have an active Tenable account to download updated software. Software for all Tenable components, including Nessus and Tenable.sc, can be downloaded from https://www.tenable.com/downloads. Follow the directions on these pages to upgrade Tenable.sc and upgrade Nessus.

Note that while Nessus plugins are updated automatically without user intervention, there is an option to manually update them. Keeping plugins up-to-date allows Tenable to identify all of the latest vulnerabilities.

2.3 Tenable.io

Tenable.io is a cloud-based platform that organizations can use to perform vulnerability scanning and reporting for their on-premises and cloud-based endpoints. In our build we used Tenable.io to provide container security for a CentOS 7 VM running Red Hat's OpenShift container orchestration software.

The platform system requirements for endpoints to run the Container Security (CS) Scanner software can be found here.

2.3.1 Tenable.io Configuration

Tenable.io is operated using an online portal. It provides a Get Started page that walks administrators through initial setup steps, such as configuring scans and linking a Nessus scanner. These steps were not needed to perform the capabilities implemented in the lab demonstration.

Administrators will need to speak with their Tenable representative to ensure access to the CS dashboard before continuing. Without access to this dashboard, they will not be able to add a connector to upload registry images or review the results from completed scans.

2.3.2 Performing Container Scans

Container registry users need to perform the following high-level steps in order to begin running container scans. For more information on getting started running the CS Scanner, please consult the following page.

1. Download and install the CS Scanner Docker image from the Tenable.io Portal. During download, you will be presented with a username and password. Please make note of them, as they will be needed during the installation.
2. Generate API keys. API keys will be needed in order for the CS Scanner tool to securely interact with and upload data to Tenable.io.
3. Set environmental variables. The following environmental variables were created and exported:
    a. TENABLE_ACCESS_KEY – This was created in step 2. It is used to allow the container security tool to connect with Tenable.io.
    b. TENABLE_SECRET_KEY – This was generated during the API key creation process. It is used to allow the tool to connect with Tenable.io.
    c. IMPORT_REPO_NAME – This is the name of the repository that you would like to export. Note that this name is what will appear in the container security dashboard of Tenable.io.
    d. REGISTRY_URI – This is the URI of the registry that you would like to import.
    e. REGISTRY_USERNAME – This is a machine account on the system that contains the correct privileges to read from the registry.
    f. REGISTRY_PASSWORD – This is the password for the account that will read from the registry.

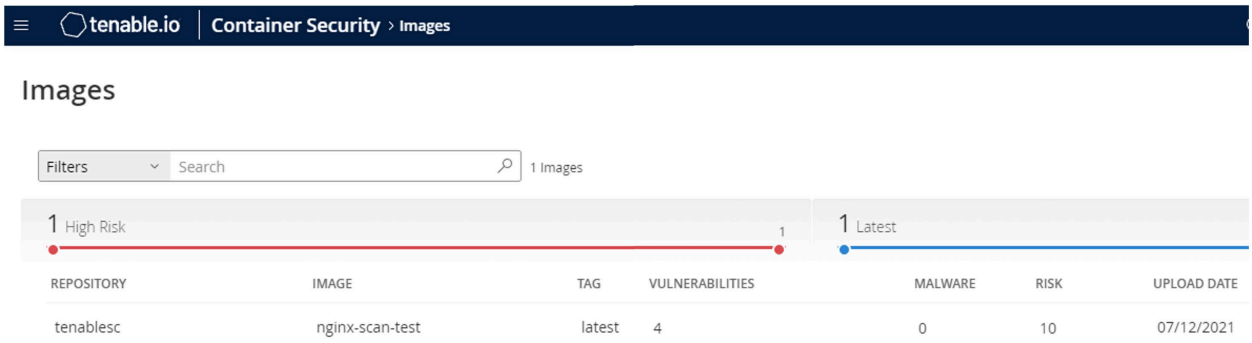
g. IMPORT_INTERVAL_MINUTES – This is how often you want the Tenable.io scanner to import and scan images. The lab implementation configured the scan to run every 1440 minutes. The scan by default will run in a manual, ad-hoc manner.

4. Configure and run the Tenable.io CS Scanner. This involves running a docker command with the environmental variables that were previously set, then importing the registry. The registry is automatically imported after a one-line command is run, without further interaction from the user.

2.3.3 Container Scan Results

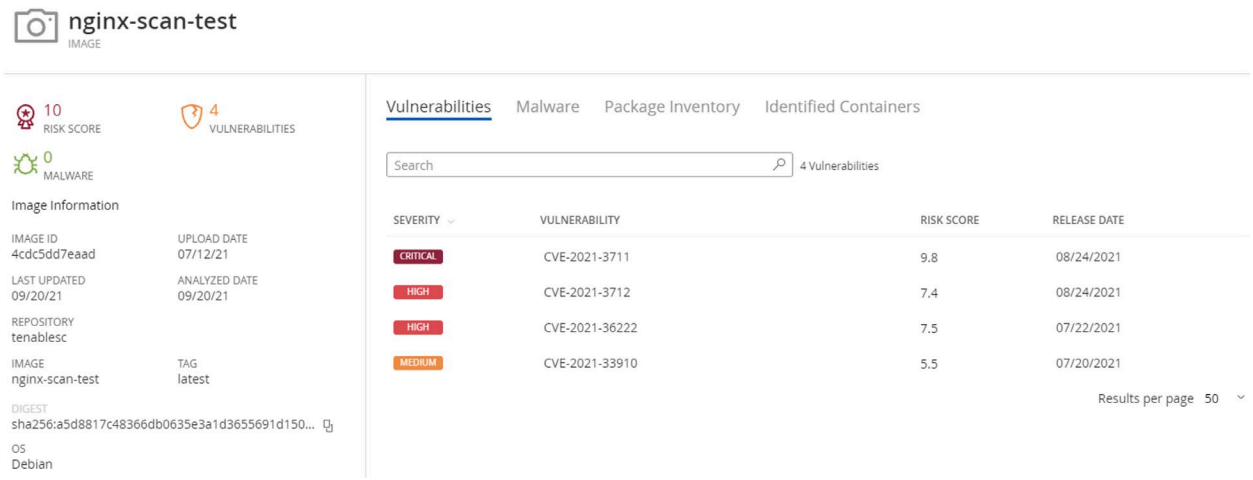
After performing the scan from Section 2.3.2, the container image data will populate inside of Tenable.io. To view scan results, a user logs in to Tenable.io and navigates to **Menu** > **Container Security** > **Images** tab. This tab presents the user with the repository and image name, the associated number of vulnerabilities or malware, risk score, and date of upload, as Figure 2-6 depicts.

Figure 2-6 Example of Container Image Data



The scan results can be further drilled into by clicking on the repository that you would like additional information on. Under this new view, administrators can see the actual vulnerabilities and CVE scores associated with containers as well as malware, package inventory, and identified containers. Figure 2-7 shows a view of the vulnerabilities associated with the lab instance's uploaded registry.

776 **Figure 2-7 Example of Container Vulnerability Information**



777 ### 2.3.4 Tenable.io Maintenance

778 Tenable.io is a SaaS offering with updates automatically provided and installed by Tenable, who
779 maintains the platform.

780 # 3 Eclypsium

781 Eclypsium provides monitoring and alerting for software and hardware components for an enterprise,
782 along with advanced capabilities such as firmware integrity checking and updating. This section provides
783 information on Eclypsium installation and usage. In this build, we utilized Eclypsium to provide agent-
784 based identification of hardware and firmware for our laptop, desktop, and server endpoints while also
785 monitoring the firmware for vulnerable or end-of-life versions. We utilized both the on-premises and
786 cloud-hosted versions of Eclypsium. Both solutions offered the same experience, with the cloud product
787 receiving updates faster and automatically.

788 ## 3.1 Eclypsium Installation and Configuration

789 Two machines were required for the on-premises installation: one for the main console and database,
790 and the other for data processing. The console machine should be accessible by a fully qualified domain
791 name (FQDN) DNS entry. The steps below are a basic overview of the installation. You will receive an
792 installation guide from your Eclypsium representative with more detailed instructions.

793 1. Provision two machines that meet or exceed the hardware requirements in the installation
794 guide.

795 2. Download the Eclypsium installation script and your license to the same folder.

3. Perform the installation.
4. Install Transport Layer Security (TLS) certificates by copying the private key, public TLS certificate, and the full certificate chain to the */opt/eclypisum/certs* directory. The TLS certificate was generated and signed by our internal Lab certificate authority (CA).

The SaaS version of Eclypsium comes fully provisioned and installed.

3.2 Eclypsium Scanning

Eclypsium scanning is agent-based, so the binary must be downloaded and installed on the target machine and registered to the Eclypsium before scanning can begin. To download the Eclypsium agent go to **Deployment** > **Download** to find the binary for your chosen computing platform. Eclypsium supports installer binaries for Windows, Windows Server, macOS, and Debian or RPM Package Manager (RPM) based linux systems. You must also use an access token (a random character string) for the registration. This token is used both to ensure that only desired endpoints are registered, and optionally to register devices in groups depending on the token used. Device tokens can be managed by navigating to **Administration** > **Tokens**.

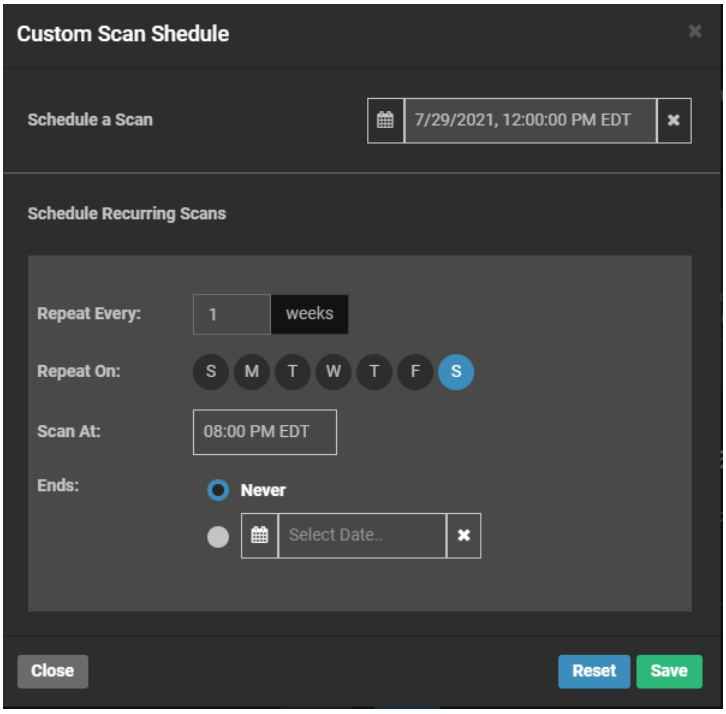
After downloading the binary onto an endpoint and generating a host registration token, the following commands were run, as an example on a CentOS 7 machine, to install the application and register the host with the console:

```
yum install eclypsium*.rpm
EclypsiumApp -s2 <DOMAIN> <REGISTRATION_TOKEN>
```

To launch an ad-hoc or manual scan, navigate to **Devices** > **Device List** and click the **Scan** button.

To schedule a recurring scan, perform the following steps:

1. Navigate to **Settings** > **Scan**.
2. Click **Schedule** under the **Scan Schedule** field.
3. Fill out the **Custom Scan Schedule** box with the information shown below.



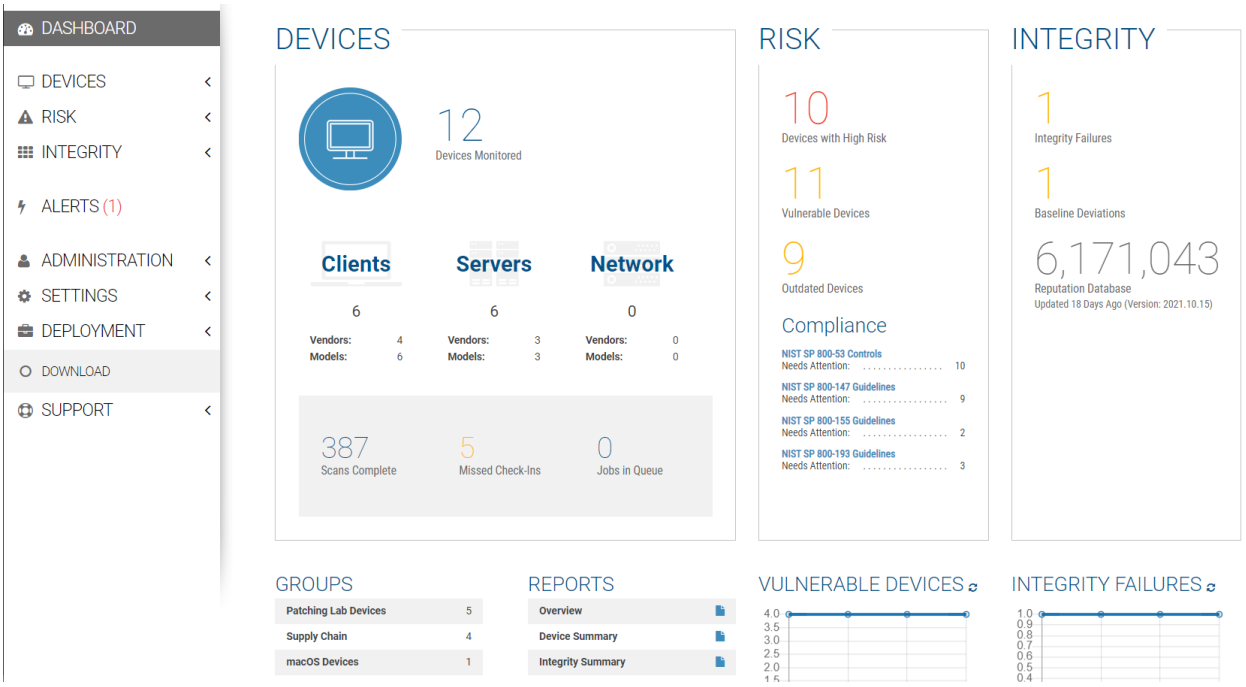
4. Click **Save**.

The above options create a scan that will run weekly on Saturdays at 8 p.m. ET. The scan schedule can be changed so that scans run more than once per week by selecting additional days, or be repeated at a different weekly interval by changing the **Repeat Every** field.

3.3 Eclypsium Reporting

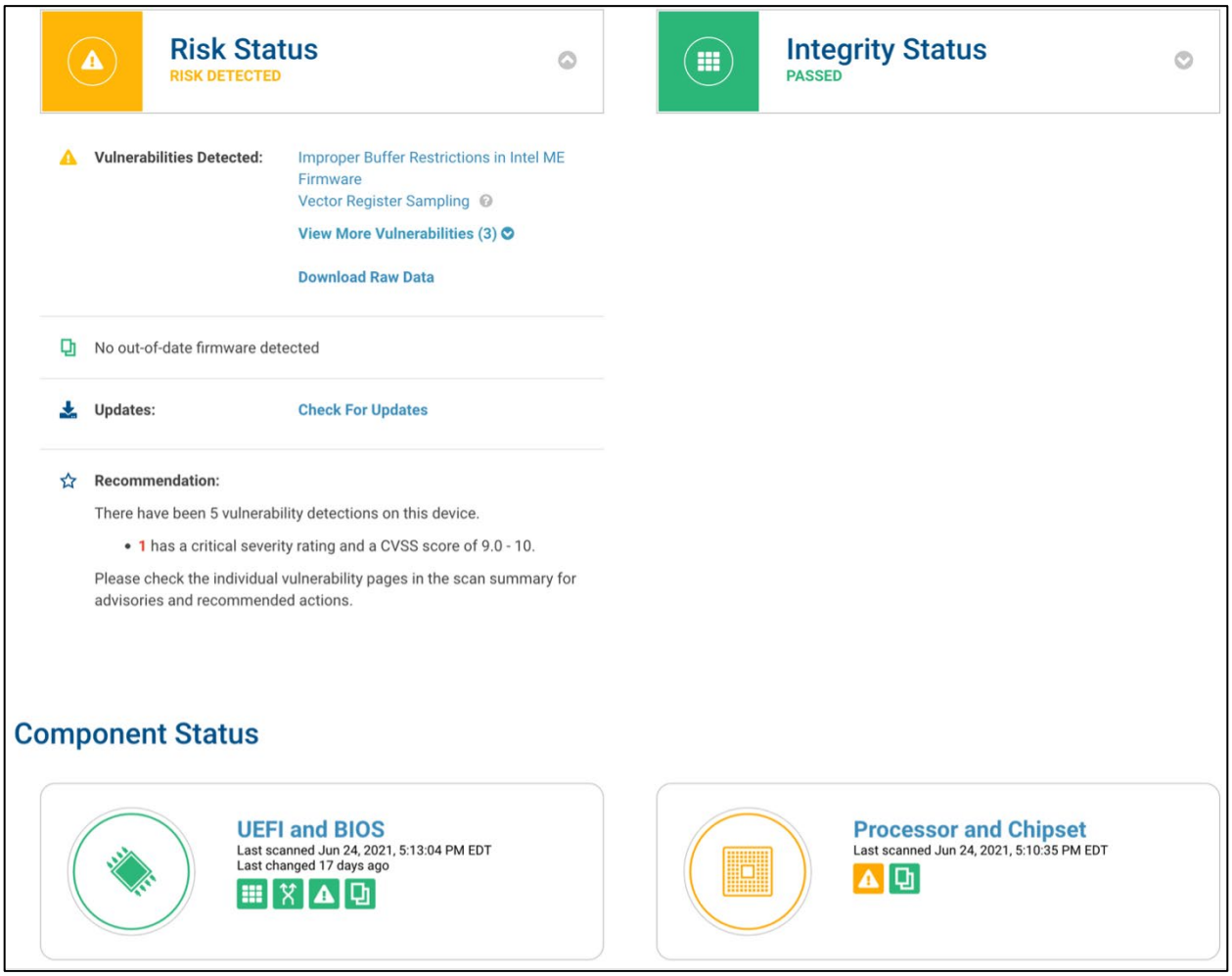
Eclypsium's main dashboard (Figure 3-1) provided firmware assessment capabilities to the build. The main dashboard provided a quick view into monitored devices, devices at risk, and the integrity of installed firmware. The **Devices** pane displayed information on the devices that were actively being monitored by the Eclypsium agent and presented that information grouped by device type (Clients, Servers, Network). The **Risk** pane displayed information regarding systems that were affected by vulnerable firmware versions with high CVSS scores. The **Risk** pane also showed all vulnerable devices and devices that were running outdated firmware. The **Integrity** pane showed devices with integrity failures and baseline deviations. Eclypsium keeps a running database of good firmware hashes to compare to an installed firmware hash to check for malicious or potentially compromised firmware versions.

836 **Figure 3-1 Eclypsium Main Dashboard**



837 Figure 3-2 provides an example of details found on a scanned device. The device registration steps were
838 performed, and a scan was conducted automatically. Although there were vulnerabilities found in the
839 chipset firmware, Eclypsium determined that no updates were available. Additionally, Eclypsium
840 provided vulnerability and integrity information for device components such as the CPU, Basic
841 Input/Output System (BIOS), and Peripheral Component Interconnect (PCI) devices; this was outside the
842 scope of this project.

843 **Figure 3-2 Eclypsium Dashboard Device Details**



844 ## 3.4 Updating Firmware

845 There is an update script from Eclypsium for automatically finding firmware updates for endpoints. The
846 script downloads the new firmware, and then the administrator performs the update manually with the
847 downloaded file. After obtaining the script (currently a python file) from Eclypsium, follow these steps:

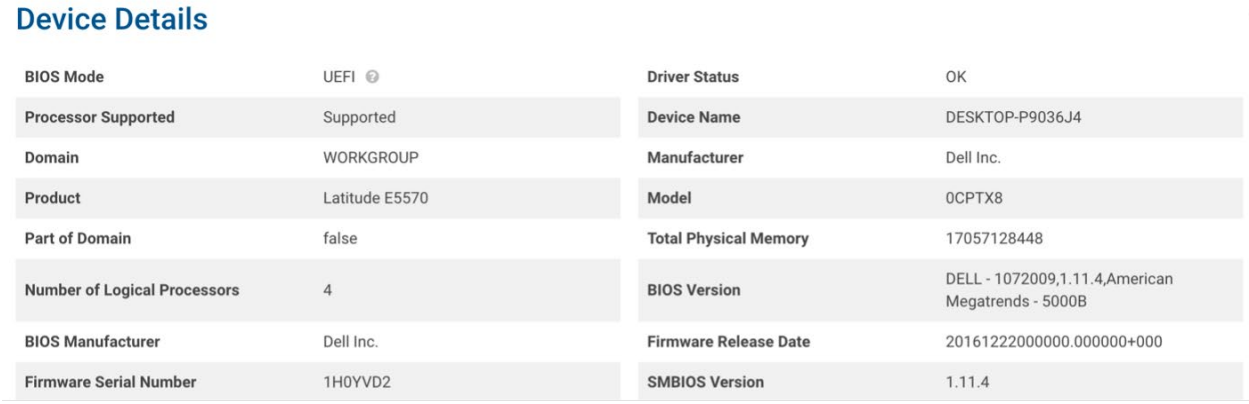
848 1. Ensure the endpoint you want to update the firmware on has the required python dependencies
849 installed so it will be able to execute the script.

850 2. Put the script on the machine and run it. It will automatically find and download the latest
851 firmware update file.

852 3. Run the downloaded file to update the firmware.

Figure 3-3 and Figure 3-4 show the characteristics of a System Management BIOS (SMBIOS) before and after running the Eclypsium firmware update script. Note that the SMBIOS Version has changed from 1.11.4 to 1.22.3 after running the update script and manually installing the downloaded firmware binary.

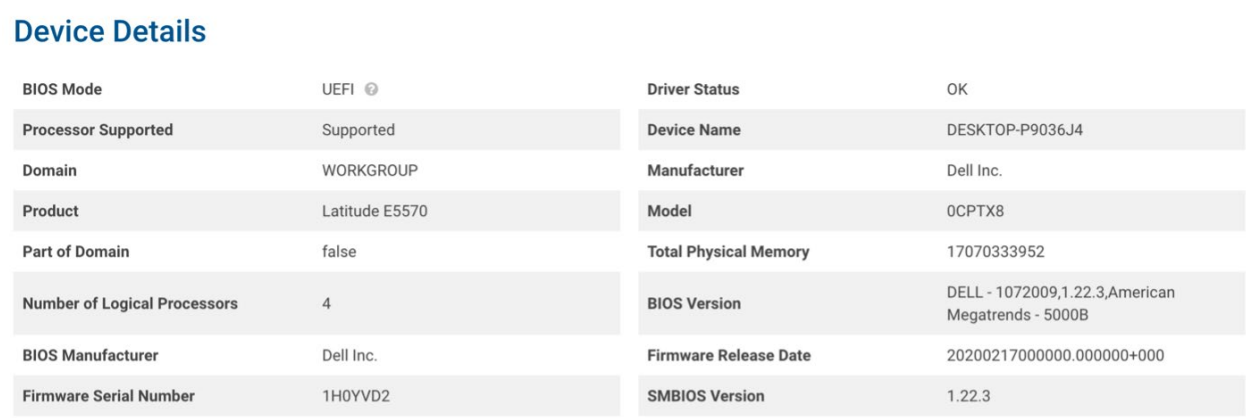
Figure 3-3 SMBIOS Before Eclypsium Firmware Update Script



Device Details

BIOS Mode	UEFI	Driver Status	OK
Processor Supported	Supported	Device Name	DESKTOP-P9036J4
Domain	WORKGROUP	Manufacturer	Dell Inc.
Product	Latitude E5570	Model	0CPTX8
Part of Domain	false	Total Physical Memory	17057128448
Number of Logical Processors	4	BIOS Version	DELL - 1072009,1.11.4,American Megatrends - 5000B
BIOS Manufacturer	Dell Inc.	Firmware Release Date	20161222000000.000000+000
Firmware Serial Number	1H0YVD2	SMBIOS Version	1.11.4

Figure 3-4 SMBIOS After Eclypsium Firmware Update Script



Device Details

BIOS Mode	UEFI	Driver Status	OK
Processor Supported	Supported	Device Name	DESKTOP-P9036J4
Domain	WORKGROUP	Manufacturer	Dell Inc.
Product	Latitude E5570	Model	0CPTX8
Part of Domain	false	Total Physical Memory	17070333952
Number of Logical Processors	4	BIOS Version	DELL - 1072009,1.22.3,American Megatrends - 5000B
BIOS Manufacturer	Dell Inc.	Firmware Release Date	20200217000000.000000+000
Firmware Serial Number	1H0YVD2	SMBIOS Version	1.22.3

3.5 Updating Eclypsium

The Eclypsium on-premises upgrade process required downloading a script and running it in the same folder Eclypsium was installed in. Our experience with updating Eclypsium was a successful one-step process. After running the script and restarting the Eclypsium service, the dashboard was updated. Eclypsium provides materials to customers on how to update their on-premises installations.

The cloud-hosted version of Eclypsium updates automatically, with no user interaction required. The on-premises version of Eclypsium is not updated automatically because it is tied closely to environment

policies. Eclypsium users will receive a notification on the main console screen when updates are available. Managed endpoints can be configured to automatically update the installed endpoint driver or update manually if needed.

4 VMware

In our build we used VMware vRealize Automation SaltStack Config 8.3.0 to provide configuration management, vulnerability management, and patch deployment. SaltStack Config was used to manage Windows workstations and servers, a macOS laptop, and Linux/Unix based VMs and servers. A full list of OSes that SaltStack Config can manage can be found here.

VMware vRealize Automation SaltStack Config is deployed with a "Salt master" server that manages endpoints via an installed agent referred to as the "Salt minion." In the build, the following SaltStack Config server components were deployed on a single VM running CentOS 7:

- **Salt master**: The Salt master service provided the main connection between SaltStack Config and the targeted endpoints running the minion agent. The Salt master plugin also communicated with the backend PostgreSQL database to access stored jobs and job configuration files.
- **Returner as a Service (RaaS)**: RaaS provided the communication between the SaltStack Config web user interface and connected Salt master nodes.
- **PostgreSQL database**: RaaS used a PostgreSQL database to store minion data, the output from job returns, event data, files, local user accounts, and settings for the user interface.
- **Redis database**: RaaS used a Redis database for temporary storage for items such as cached data. It also used this database to hold queued work for deployment.

4.1 VMware vRealize Automation SaltStack Config Installation and Configuration

VMware vRealize Automation SaltStack Config and its components listed above were installed via the SaltStack installer script on a CentOS 7 VM, with hardware details included in Section 1.4. SaltStack Config has the following software dependencies:

- OpenSSL
- Extra Packages for Enterprise Linux (EPEL)
- Python cryptography
- Python OpenSSL library

More information on SaltStack Config requirements can be found here.

The SaltStack Config installation process consists of the following steps:

1. Obtain the SaltStack Config installer zip file from your SaltStack representative.
2. Unzip the zip file on the desired installation node.
3. Run the *setup_single_node.sh* script.
4. Allow port 443 access for reaching the SaltStack Admin Web graphical user interface (GUI).
5. Allow port 4505 and 4506 access for communication between the Salt master and minion agents.
6. Install the license key.

More information on installing SaltStack Config can be found here.

4.2 Salt Minion Agent

The Salt minion agent is how SaltStack Config communicates with endpoints to perform configuration. The minion agent needs to be installed on any endpoints that will be managed by SaltStack Config. The minion agent is available for various OSs and can be found here along with OS-specific installation instructions.

The minion agent can be installed and configured with the following steps:

1. Download and install the minion agent.
2. Edit the minion agent with the IP address of the Salt master server. Note that by default, the minion will use the DNS name of 'salt' when trying to connect to the Salt master server. On Linux-based systems the configuration file located under */etc/salt/minion* can be edited to use custom IP addresses or hostnames instead. On Windows-based systems, this information can be edited using the minion configuration wizard.
3. Start the minion agent.
4. Accept the minion key.

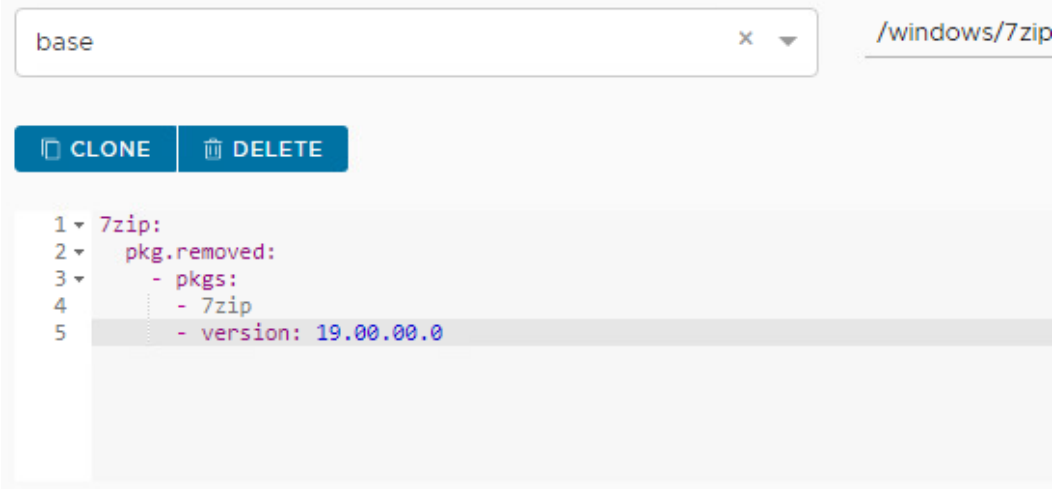
The Salt minion agent uses a public/private key pairing for communicating with the SaltStack Config server. The key generation process takes place automatically on the client system, and the minion public key is automatically sent to the Salt master server. The public key of the minion agent will need to be accepted on the Salt master server so that secure communication can take place. Steps for accepting a new minion key can be found here. Note that jobs will not be able to be issued to endpoints unless the minion key is accepted in the SaltStack Config console.

4.3 SaltStack Config Jobs

SaltStack Config uses jobs to run remote execution tasks on endpoints. The build utilized these jobs to provide configuration management capabilities. Jobs were created, scheduled, and executed via the SaltStack Config web console.

For brevity, and because jobs are highly customizable, this guide includes one example of creating and running a job. The example job demonstrates removing 7zip version 19 from a Windows endpoint in an emergency workaround scenario, where an administrator chooses to remove a vulnerable product that cannot be patched. The following are the steps used in the build to set up and execute this job:

1. Click on **Config** > **File Server**.
2. Click on **base** from the **saltenv** dropdown menu. Base corresponds to one of the default file directories that are created to hold configuration files.
3. Type *windows/7zip.sls* for the path name.
4. In the field name below, add the information in the screenshot, then click **SAVE**.



5. Next, click **Config** > **Jobs**, then click **Create Job**. Edit the fields listed in Table 4-1 so they have the specified values.

Table 4-1 Specified Values for Creating "Uninstall 7zip" Job Using SaltStack Config

Field	Value	Explanation
Name	Uninstall 7zip	This is the name of the job.
Command	Salt	The salt command allows for all salt functions to be loaded and available for choosing.

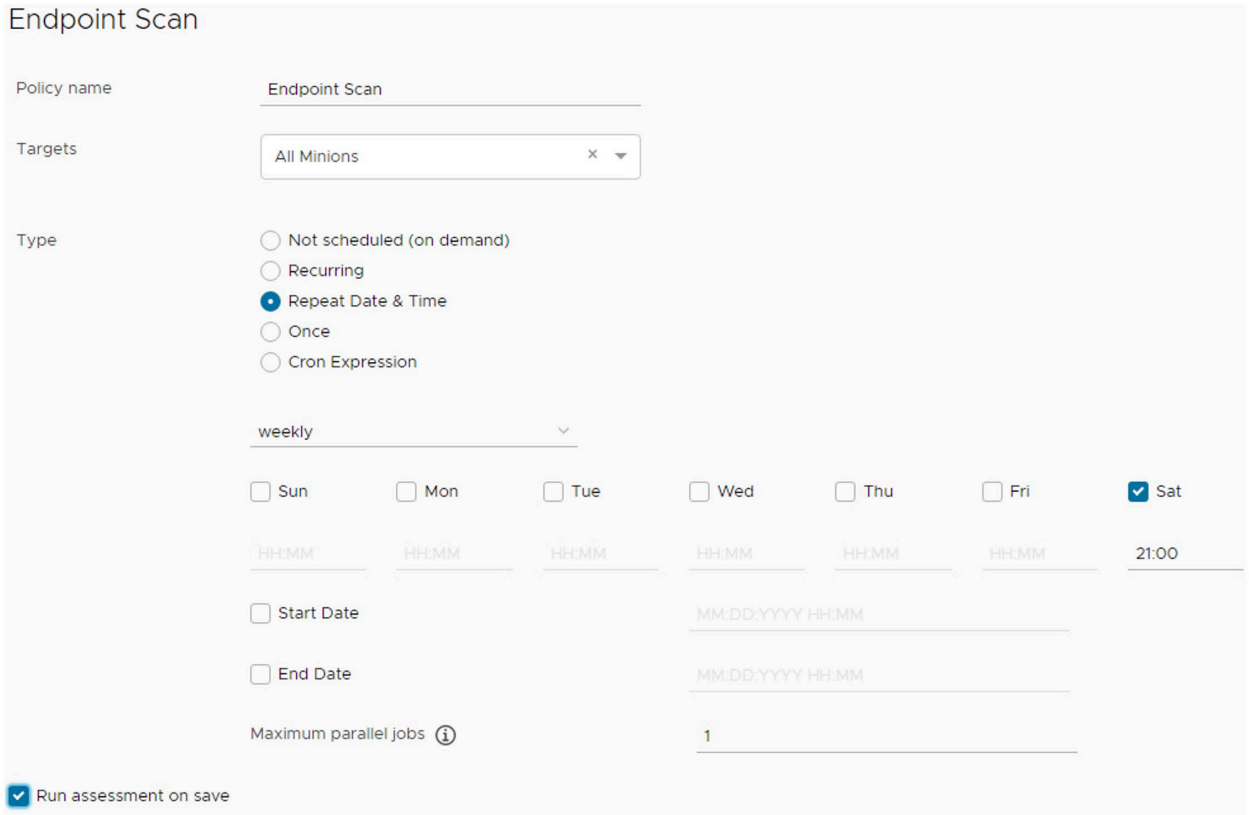
Field	Value	Explanation
Targets	Windows	This field allows for different groups of machines to have configurations applied to them. The default way that SaltStack groups machines is by OS; however, other target groups can be created based on device attributes.
Function	state.apply	The state.apply function allows for custom state files or .sls configuration files to be applied to an endpoint.
Environments	Base	Base corresponds to one of the default file directories that are created to hold configuration files.
States	windows.7zip	The states field corresponds to the file with the configurations that are to be pushed down to the endpoint. In this example, this corresponds to the uninstallation of 7zip configuration file.

6. Click on **Minions**, then select the **Windows Target Group**.
7. Click **Run Job**. Under the **Job** dropdown menu, select **Uninstall 7zip**.
8. Select **Run Now**.

4.4 SaltStack SecOps

SaltStack SecOps, an add-on component for vRealize Automation SaltStack Config, was utilized to provide vulnerability and patch management capabilities. SaltStack SecOps can be configured to run scheduled assessments of endpoint vulnerabilities with the following steps:

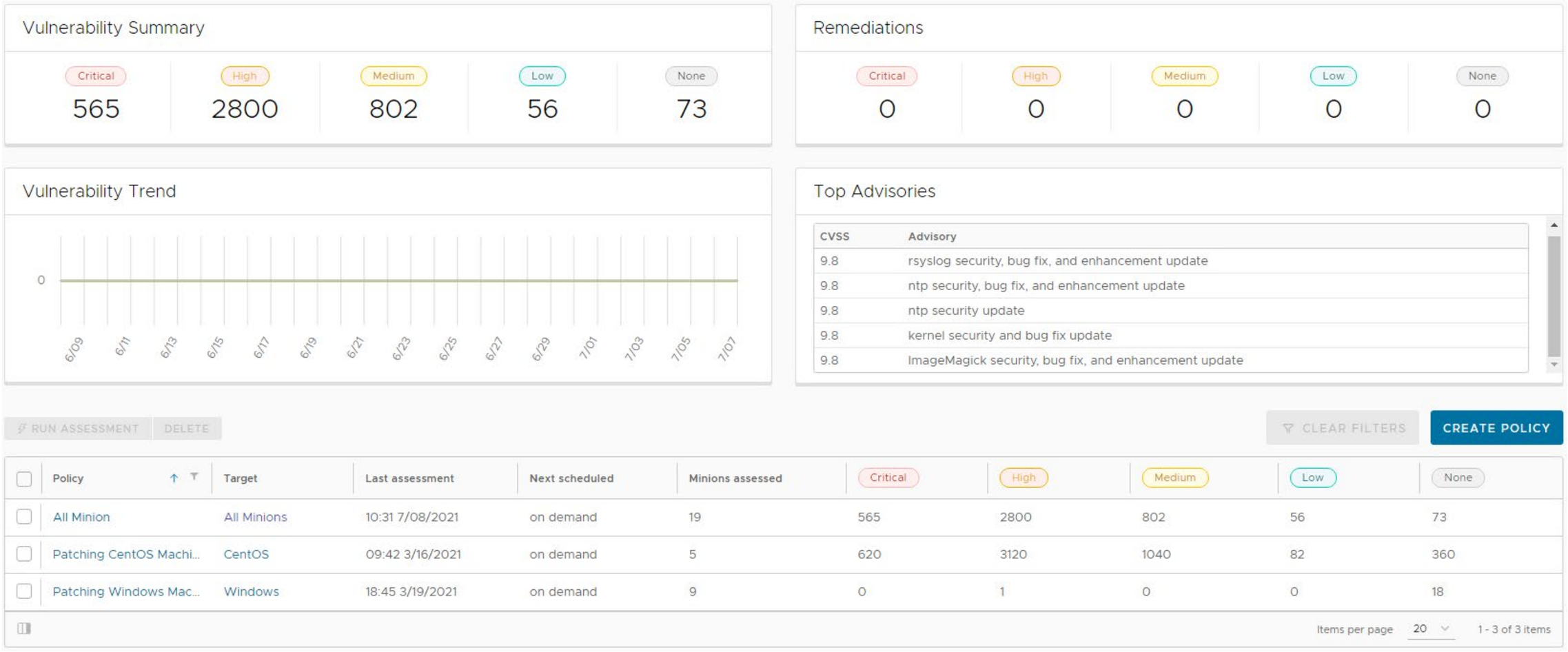
1. Click on **Protect** > **Policies** under the SaltStack Config Web GUI.
2. Click **Create Policy**.
3. Type in **Endpoint Scan**.
4. Under **Targets**, select **All Minions**. This performs a scan of all connected network endpoints regardless of OS. A scan targeting a specific OS or other defined target group could be performed instead by selecting a different value.
5. Under **Type**, choose **Repeat Date & Time,** and fill out the other options as shown.



6. Make sure that **Run assessment on save** is checked.
7. Click **Save**. The above scan will automatically run and be scheduled to run weekly on Saturdays at 9 p.m. without further user interaction.

After running the scan, the Vulnerability Summary and Top Advisories dashboard begins to populate, as captured in Figure 4-1. The image shows that the SaltStack SecOps engine has started collecting vulnerability information and categorizing it by severity level. The Top Advisories dashboard shows vulnerabilities detected in the scan that have the highest CVSS score. In the scan, the top advisories all have scores of 9.8.

966 **Figure 4-1 SaltStack SecOps Vulnerability Summary and Top Advisories Dashboard**



SaltStack SecOps can also be used to remediate endpoints. To do so, follow these steps:

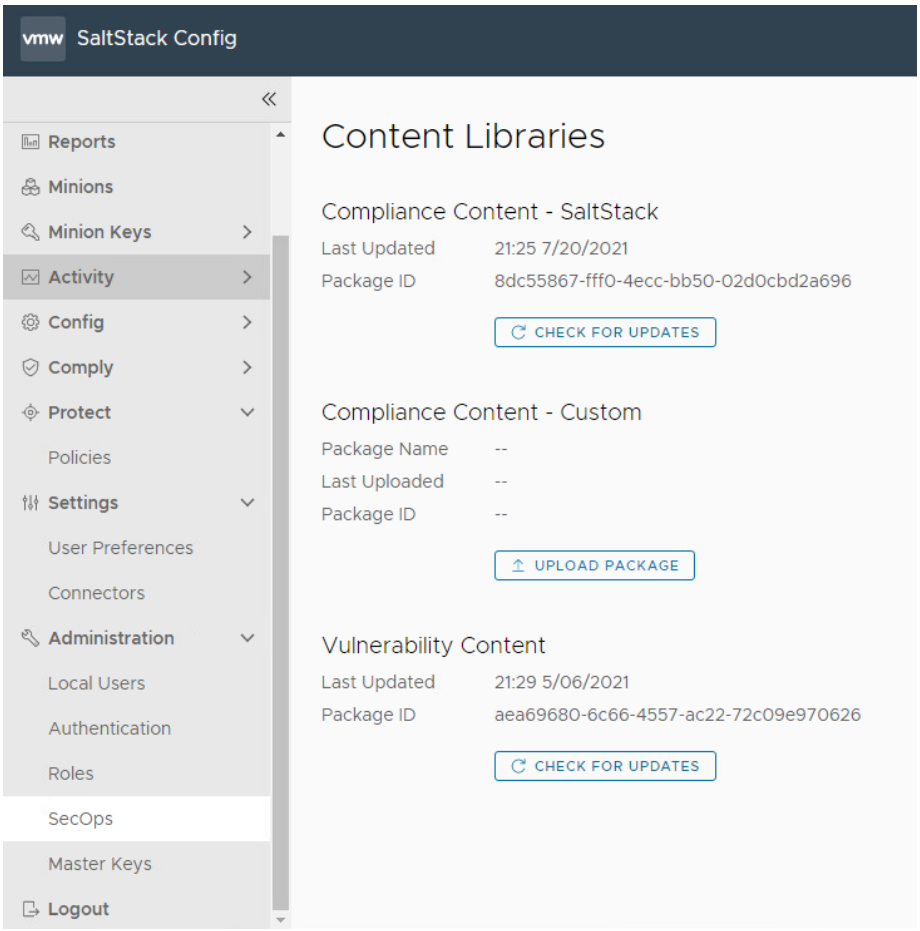
1. Click the "Endpoint Scan" policy that was created previously.
2. From the resulting list, either single remediations can be selected, or you can choose to select all remediations.
3. When the desired patches are selected, click **Remediate**.

4.5 vRealize Automation SaltStack Config Maintenance

All SaltStack Config components should be kept up to date. You are required to have an active VMWare account to download updated software. Software for all SaltStack Config components can be downloaded from here. To upgrade SaltStack Config, follow the directions in Section 10 (Upgrade from a previous version) of *Installing and Configuring SaltStack Config*.

SaltStack vulnerability data is kept up to date automatically without user interaction. To perform a manual check for updates, perform the following steps:

1. Log in to the SaltStack web console.
2. Navigate to **Administration** > **SecOps**.
3. Click **CHECK FOR UPDATES** under the **Vulnerability Content** section.



5 Cisco

In this implementation, we used the Cisco Firepower Threat Defense (FTD) firewall to provide network access management capabilities and Cisco Identity Services Engine (ISE) to provide device discovery capabilities. The Cisco Firepower Management Center (FMC) product was utilized to manage Cisco FTD. All Cisco products in the build were virtual appliances that were deployed in VMWare ESX via Open Virtualization Formats (OVFs) downloaded from the Cisco website.

5.1 Cisco Firepower Threat Defense and Firepower Management Center

Cisco FTD is a next-generation virtual firewall that was used to provide networking to the patching architecture. The build utilized Cisco FTD 6.4.0 to enforce network access control using firewall rules. Cisco FTD was deployed and managed in the lab via a separate Cisco FMC VM. This section walks through installing and configuring Cisco FTD and Cisco FMC.

5.1.1 Cisco Firepower Management Center Installation

Cisco FMC was utilized to manage an instance of Cisco FTD. With this in mind, it is suggested to set up FMC first. Installing and setting up the FMC virtual appliance involved the following steps:

1. Download the FMC VM tar file from the Cisco Downloads page. Note that you will need a Cisco account to download it.
2. Deploy the OVF in VMWare.
3. Perform initial configuration of the FMC. This included tasks like accepting the End User License Agreement (EULA), setting a password, and configuring network settings.

5.1.2 Cisco Firepower Threat Defense Installation

For our build, installing the Cisco FTD VM consisted of the following steps:

1. Download the OVF from the Cisco Downloads page.
2. Deploy the Cisco FTD VM using the VMware vSphere web client.
3. Complete the Cisco FTD VM setup using the command line interface (CLI). This included performing initial configuration, such as setting up network information, user credentials, management mode, and firewall mode. In our build, we chose **no** for "Enable Local Manager" to ensure that the FTD was managed by the FMC from Section 5.1.1. The FTD was set to routed firewall mode, which allowed for IP-based separation between subnets.
4. Register the Firepower Threat Defense to the Firepower Management Center. This included configuring network information for the management port, which was the IP address that the management center VM communicated with.

5.1.3 Licensing Cisco FTD with Cisco FMC

When first logging into the Cisco FMC, a license needs to be applied to the Cisco FTD instance. Instructions can be found here. The smart licensing feature allows for individual features to be licensed to meet organizational needs. The license types listed in Table 5-1 were applied to our build, and they granted the specified capabilities.

Table 5-1 License Types and Granted Capabilities for Cisco FTD

License Type	Granted Capabilities
Base	User and application control, switching, routing, network address translation (NAT)
Threat	Intrusion detection and prevention

Malware	Threat intelligence for detecting malware
URL Filtering	Category and reputation-based uniform resource locator (URL) filtering
AnyConnect VPN Only	Remote access virtual private network (VPN) configuration

5.1.4 Cisco FTD Initial Network Configuration

After licensing the Cisco FTD instance, the next step is to configure networking information for the firewall interfaces. Security zones need to be created; they allow firewall interfaces to be grouped together in order to apply configuration and policy. To create security zones, perform the following steps:

1. Choose **Objects** > **Object Management**.
2. Choose **Interface** from the list of object types.
3. Click **Add** > **Security Zone.**
4. Enter a name.
5. Select **Routed** from **Interface Type.**
6. Click **Save.**

The security zones described in Table 5-2 were created in support of our build:

Table 5-2 Security Zones Created for Cisco FTD

Security Zone	Zone Description
Outside Zone	Contained the wide area network (WAN) interface that sat between the firewall and the internet gateway
Endpoints	Contained the interface that communicated with all lab endpoints which represented end user devices and servers
Shared Services	Contained shared common services such as DHCP and DNS
Patching Products	Contained all deployed patching products and services

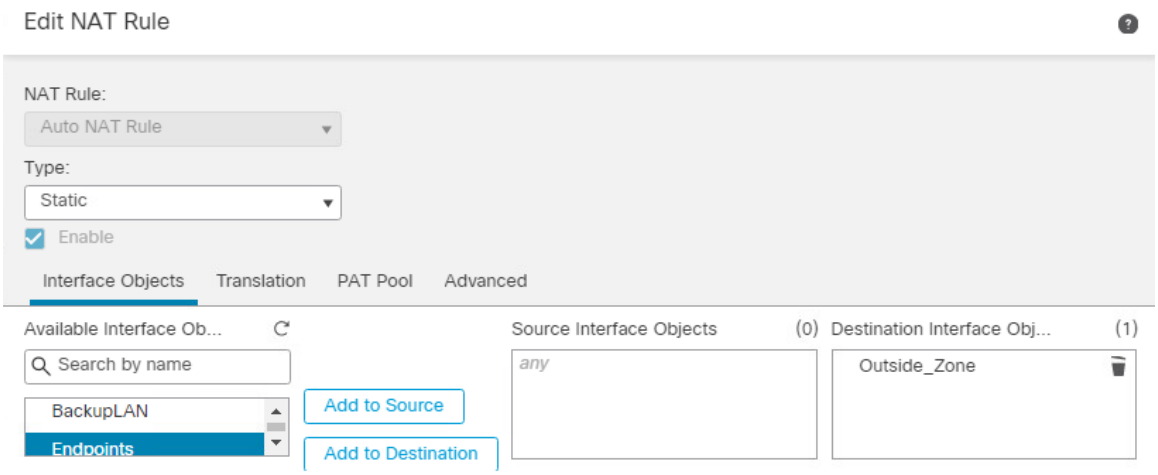
The next step is to edit each firewall interface with the correct IP address for your organization and the appropriate security zone:

1. Select **Devices** > **Device Management** and click **Edit** (✎) for your FTD device. The **Interfaces** page is selected by default.
2. Click **Edit** (✎) for the interface you want to edit.

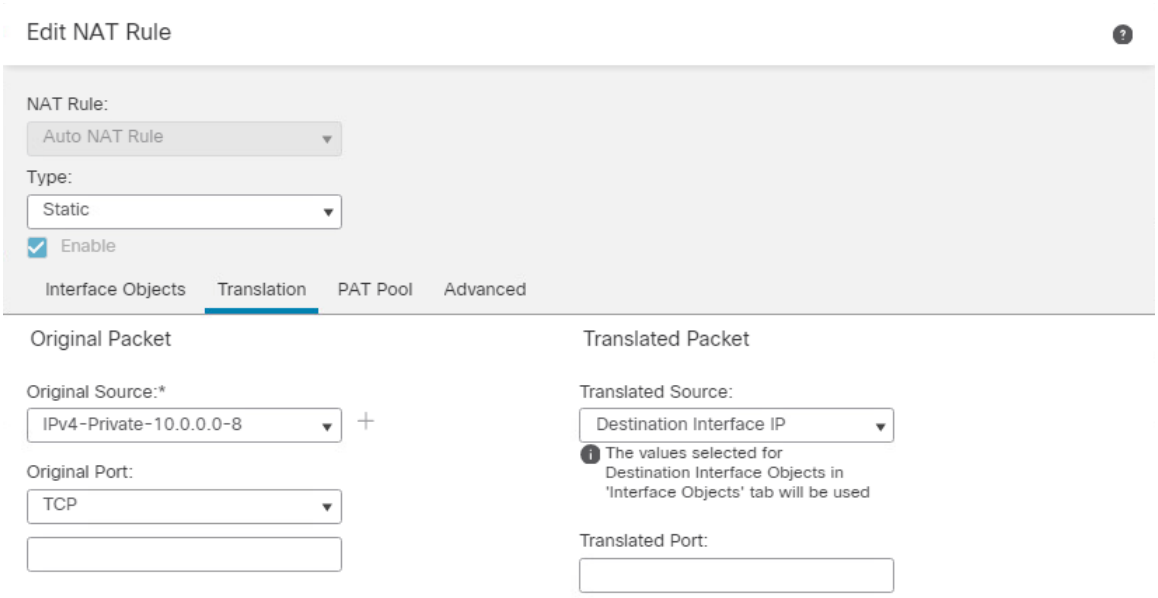
3. Enable the interface by checking the **Enabled** check box.
4. Under the **Security Zone** dropdown, select the correct security zone.
5. Under the **IPv4** tab, enter the appropriate IP address information.
6. Click **Ok**.
7. Click **Save**.

The last step is to enable NAT. Since private IP addresses cannot traverse the public internet, a NAT rule needs to be created to allow the public IP address for the firewall to be used for external network traffic from internal network endpoints using private IP addresses. To create a NAT policy, perform the following steps:

1. Select **Devices** > **NAT.**
2. Click **New Policy** > **Threat Defense NAT** to create a new policy. Give the policy a name, optionally assign devices to it, and click **Save**.
3. Click **Edit** (✎) to edit the Threat Defense NAT policy.
4. Click **Add Rule**, then select **Auto NAT Rule**.
5. Under **Interface Objects**, leave **any** under **Source Interface Objects**, and place **Outside_Zone** under **Destination Interface Objects**.



6. Under the **Translation** tab, select **IPv4-Private-10.0.0.0-8** under **Original Source**, and under **Translated Source** select **Destination Interface IP**.



7. Click **Ok**, then click **Save.**
8. Click **Deploy** > **Select Device** > **Deploy** to deploy the NAT policy.

5.2 Cisco Identity Services Engine

Cisco ISE is a network administration product that allows for enforcement of administrator-created security and access control policies. Cisco ISE captures attributes about devices, such as IP address, MAC address, and OS in order to enforce custom policies. Cisco ISE can be deployed as a standalone system or as a primary and secondary node for high-availability deployments. Our build utilized a single ISE VM node set in standalone deployment.

5.2.1 Cisco ISE Installation

The installation process for deploying a virtualized version of Cisco ISE requires you to download the OVA from https://software.cisco.com/download/home and deploy it using VMWare. Note that you will need a Cisco account to be able to download software from Cisco. Follow the steps here for deploying the Cisco ISE OVA template.

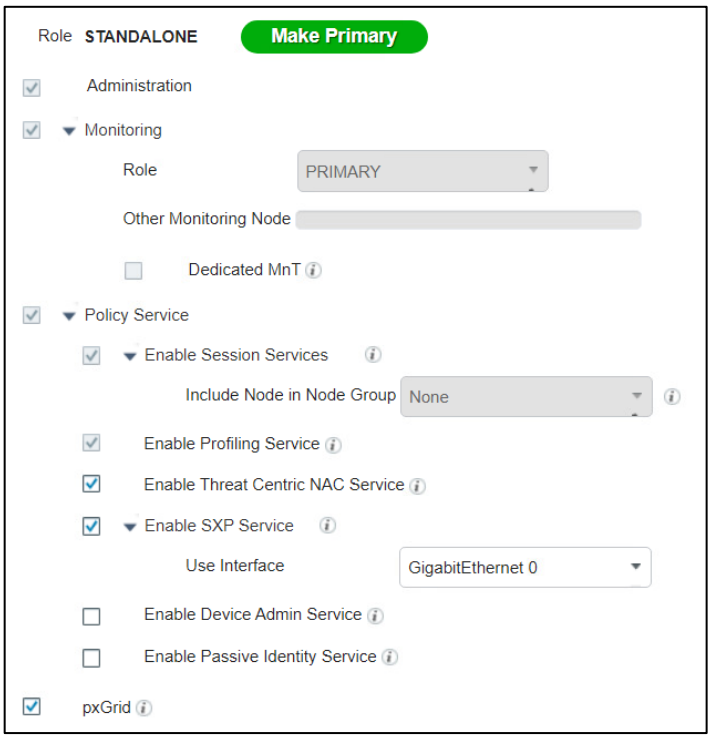
After deploying the ISE OVA, launch the VM console from VMWare. At the Cisco ISE CLI, type **setup** to start the ISE setup wizard. Use it to configure hostname and IP address information and to create admin credentials for the Web Admin portal.

Lastly, Cisco ISE needs to be licensed. Follow the guidance here to find more information on licensing your ISE deployment.

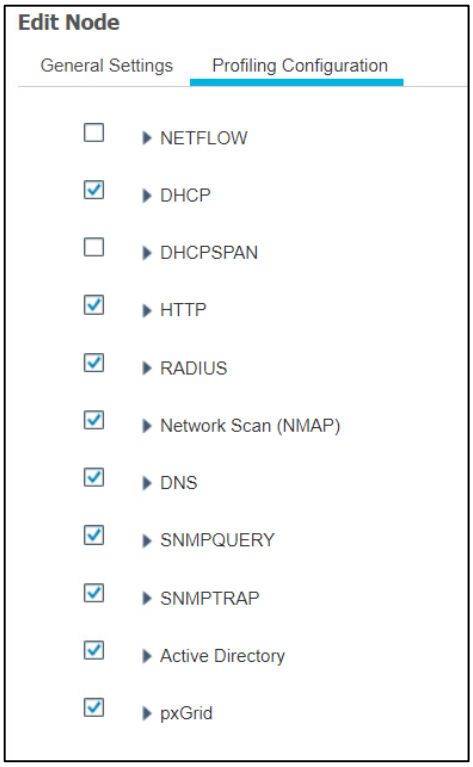
5.2.2 Cisco ISE Initial Configuration

After performing initial setup and licensing, the next step is to ensure that the Cisco ISE deployment node has the correct settings and profiling configuration services running. Perform the following steps:

1. Click **Administration** > **System** > **Deployment.**
2. Under the **General Settings** tab, ensure that the options shown below are selected.



3. Under the **Profiling Configuration** tab, ensure the following options are selected. Note that a description of the various profiling services can be found on the **Profiling Configuration** tab. When you are done selecting the options, click **Save**.



For our build, Cisco ISE needed to have an integration with AD services to perform authentication of endpoint users to the network. Cisco ISE used AD as a trusted store to authenticate users and machines to the network. To perform the integration between Cisco ISE and AD, follow the guidance here.

5.2.3 Configuring AnyConnect VPN Using Cisco FTD and Cisco ISE

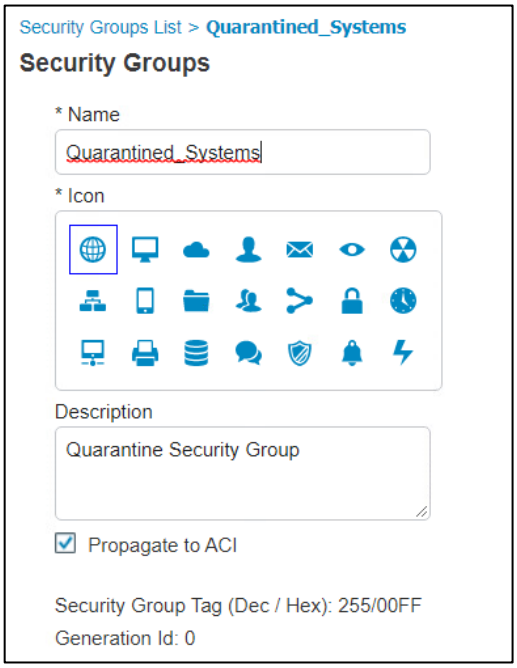
By default Cisco ISE cannot make any policy enforcement actions for devices that are not actively authenticated against it. This means that devices that are not using 802.1X authentication or the AnyConnect VPN client will not have full device attributes collected nor be subject to ISE policy rulesets. Our build utilized AnyConnect VPN integration between the Cisco FTD and Cisco ISE to demonstrate authenticating two hosts to Cisco ISE. The example assets chosen to be connected to the VPN were a Windows 10 and CentOS 7 VM. Please follow the steps here for setting up the integration.

5.2.4 Cisco Security Group Tags (SGTs)

Cisco security group tags (SGTs) are user-designated tags that can be used to group and classify devices. Each tag is then used to represent logical group privileges to inform the access policy. SGTs were used by the build to restrict access to devices that did not meet the desired organization patch level. This section covers setting up the Quarantine SGT and sharing SGTs with Cisco FTD.

First, add the Quarantine SGT to Cisco ISE with these steps:

1. Click on **Work Centers** > **Trust Sec** > **Components** > **Security Groups**.
2. Click **Add**.
3. Under **Name**, type: Quarantined_Systems.
4. Under **Description**, type: Quarantine Security Group.
5. Ensure the **Propagate to ACI** option is checked.



After adding the Quarantine SGT, it needs to be shared with the Cisco FTD. SGTs are not shared between ISE and FTD by default. ISE will have to be added as an identity source to the firewall. This communication between the firewall and ISE takes place using pxGrid. The process for setting up SGT sharing from ISE to the firewall involves:

- making sure that SGTs are published via pxGrid by Cisco ISE,
- exporting the ISE pxGrid and monitoring (MNT) system certificates for importation to FTD, and
- adding ISE as an identity source on the firewall.

The build used this integration to perform network access control on devices that were given the Quarantine SGT by ISE. This SGT was given by assessing an endpoint's current patch level. See this page for step-by-step guidance on adding Cisco ISE as an identity source.

5.2.5 Cisco ISE Integration with Tenable.sc

For our build, Cisco ISE contained an integration with Tenable.sc to perform automated scanning of endpoints as they were authenticated to ISE. ISE could then take the highest CVSS score that was associated with an endpoint and, via policy, enforce network restrictions through sharing SGTs with the Cisco firewall. The build used this capability to scan devices as they connected to the network and determine whether a quarantine action should take place.

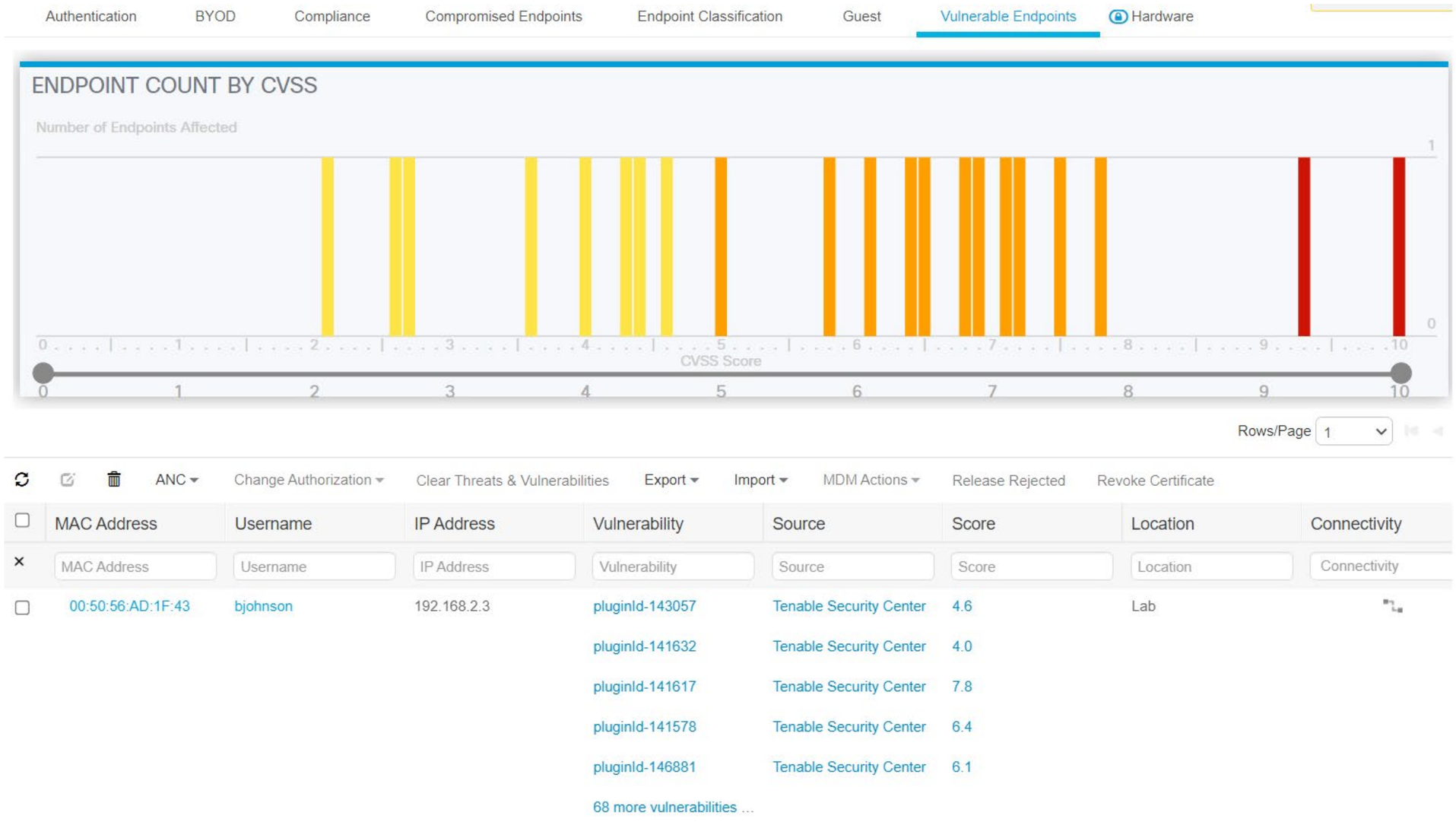
The steps for integrating ISE with Tenable.sc consist of the following:

1. Create a machine account for ISE to log in into Tenable.sc to launch a scan. The device is referred to as a machine account since it is used by a service and not a person.
2. Export the Tenable.sc Root and System certificates, and import them to Cisco ISE. This step is so there are no errors when Cisco tries to contact Tenable over Hypertext Transfer Protocol Secure (HTTPS) for API calls.
3. Configure third-party threat integrations on Cisco ISE. This will start the process of creating the integration with Tenable, including creating a Tenable adapter.
4. Configure the Tenable adapter. The adapter is how Cisco ISE will communicate with Tenable, so it needs to be configured to provide login credentials and connection options.
5. Configure an authorization profile. This configures Cisco ISE to assess vulnerabilities via the newly created Tenable adapter.

Step-by-step guidance on integrating Cisco ISE with Tenable.sc is available here. Note: Your ISE instance will need to be version 2.7 or higher.

Once the integration between Cisco ISE and Tenable is configured correctly, vulnerability data is viewable for connected endpoints. To view vulnerability data for connected devices, go to **Context Visibility** > **Endpoints** > **Vulnerable Endpoints**. The collected device information, such as the example in Figure 5-1, shows the affected IP address, current user, Tenable plugin ID, and CVSS score.

1142 **Figure 5-1 Cisco ISE View of Vulnerability Data for Connected Devices**



1143 The highest CVSS score associated with a device was utilized by the patching lab to create policy that would restrict network access to devices with a vulnerability that exceeded a CVSS threshold score of 7. This threshold was designed to
1144 block devices that have high and critical severity scores.

5.2.6 Cisco ISE Integration with Cisco Catalyst 9300 Switch

For our build, Cisco ISE contained an integration with a physical Cisco Catalyst 9300 switch located in the lab network. This allowed Cisco ISE to perform 802.1x port-based authentication for devices that were connected via ethernet. The build used this capability to authenticate devices to the network and then later scanned authenticated devices to ensure they were at the appropriate patch level. The example implementation applied 802.1x authentication to port 40 of a 48-port switch.

The following is an abbreviated version of the steps we performed in the lab to integrate ISE with the Cisco Catalyst 9300 switch. For more detailed guidance, consult the following Cisco guide.

1. Access the admin console of the Cisco switch via a physical connection or remote protocol.
2. Go to global configuration mode by typing `config t` and then enter the following:

```
aaa new-model
!
aaa group server radius ise
 server name ISE
!
aaa authentication dot1x default group ise
aaa authorization network default group ise
aaa accounting update newinfo periodic 1440
aaa accounting dot1x default start-stop group ise
!
aaa server radius dynamic-author
 client 10.132.6.12 server-key password
!
aaa session-id common
switch 1 provision c9300-48p
!
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
```

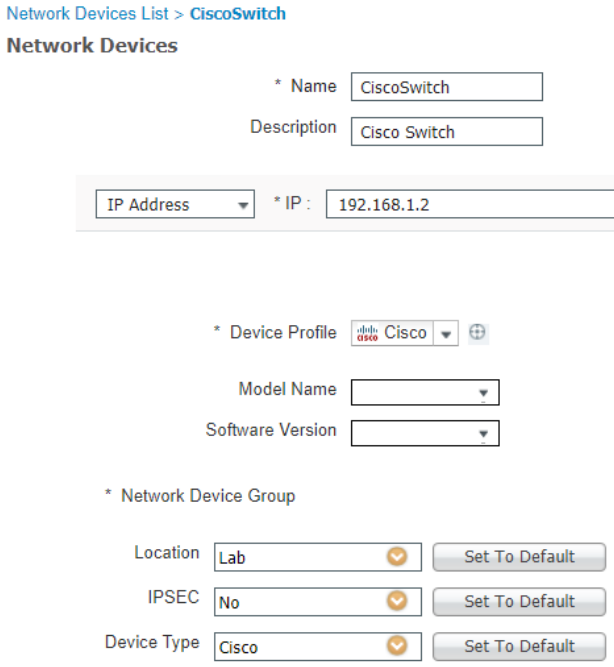
```
radius-server attribute 31 mac format ietf upper-case
radius-server attribute 31 send nas-port-detail
!
radius server ISE
 address ipv4 10.132.6.12 auth-port 1645 acct-port 1646
 key password
```

3. Configure interface 40 by typing `interface Gi10/40` at the switch terminal and then entering the following information:

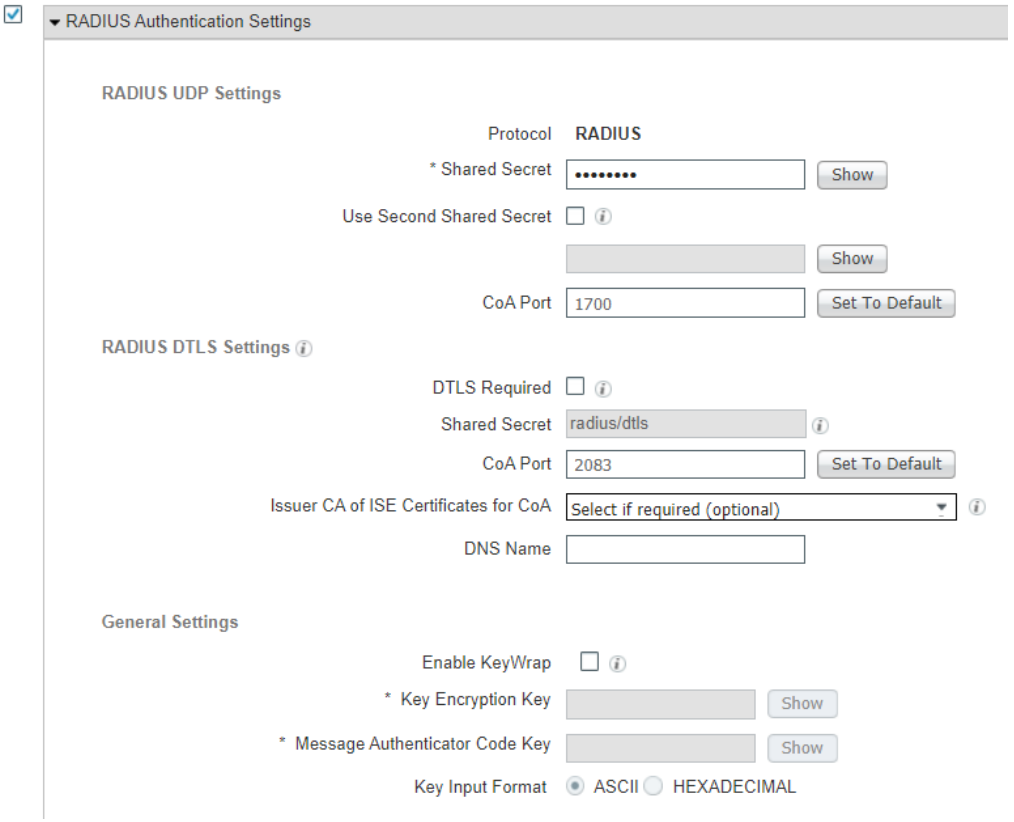
```
switchport mode access
 authentication event fail action next-method
 authentication event server dead action authorize vlan 1345
 authentication event server dead action authorize voice
 authentication event server alive action reinitialize
 authentication host-mode multi-auth
 authentication open
 authentication order dot1x mab
 authentication priority dot1x mab
 authentication port-control auto
 authentication periodic
 authentication timer reauthenticate server
 authentication violation restrict
 mab
 dot1x pae authenticator
 dot1x timeout tx-period 10
 spanning-tree portfast
```

4. Add the Cisco Switch to ISE by navigating to **Administration** > **Network Resources** > **Network Devices** and clicking the **Add** button.

5. In the **Network Devices** field, fill out the information shown below to ensure that Cisco ISE knows the IP address of the switch, network device group information, and has a name and description for the new device.

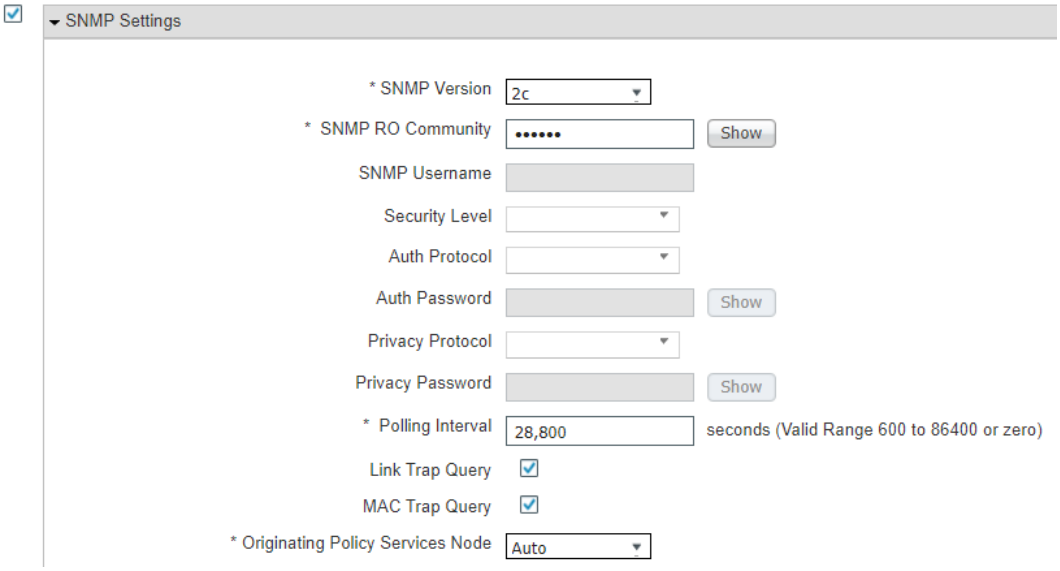


6. Ensure the **RADIUS Authentication Settings** box is checked, then fill out the information shown below. Please note that the Share Secret field corresponds with the **radius server key** field from the last line of the configuration in step 2.



1209

1210 7. Next, select the checkbox by **SNMP Settings**, and fill out the information depicted below.



1211

8. Click the **Save** button.

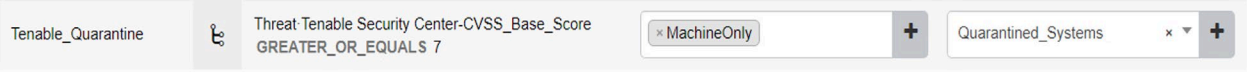
5.2.7 Cisco ISE Policy Sets

Cisco ISE policy sets are policy-based rules that are written to group devices together. Group devices can then have access control policies applied. Our build utilized policy sets to create rules that would apply network access control policies to devices that did not meet the appropriate patch level. Guidance for setting up policy sets can be found here.

5.2.7.1 VPN Policy Set

The following policy set was created for the build to enforce network restrictions on VPN devices that did not meet the desired patching threshold. As a reminder, VPN devices were chosen because network enforcement can only be performed on actively authenticated devices. The following steps walk through setting up the patching example policy set:

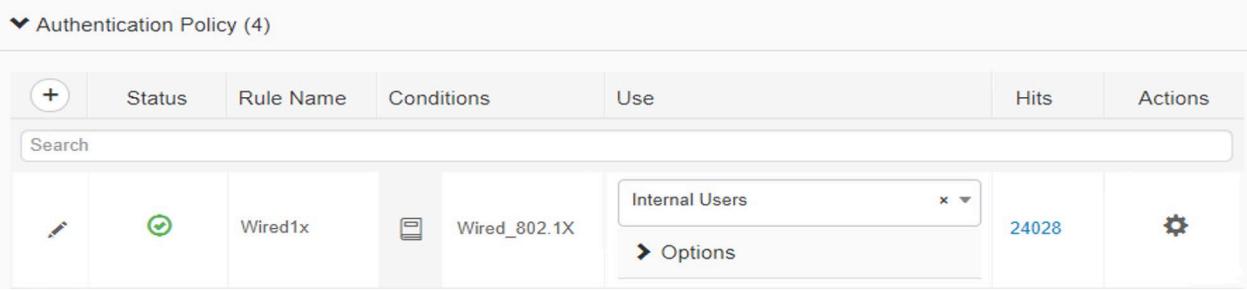
1. In the Cisco ISE Web Console, click **Policy** > **Policy Sets**.
2. Click the Add icon.
3. Under the **Policy Set Name** field, type: VPN.
4. Click the plus button under the conditions field.
5. In the **Editor** field, select **Click to add an attribute field**.
6. Click the Network Device Button  .
7. Click the **Device Type** attribute.
8. Under the **Choose from list or type** dropdown, select **All Device Types#VPNDevice**.
9. Click the **Use** button.
10. Click the arrow under **View** on the newly created VPN Policy.
11. Under the **Authorization Policy – Global Exceptions** tab, add the rule depicted below. It indicates that if an endpoint has a vulnerability with a CVSS score greater than 7, the device receives the Quarantined_Systems security group tag. This rule was placed into the **Global Exceptions** tab because it allows these rules to be checked first. This is important, as it allows rules in this category to override any rules that may grant network access to a device.



5.2.7.2 Wired 802.1x Policy Set

The following policy set was created for the build to enforce network restrictions on wired 802.1x connected devices that did not meet the desired patching threshold. The following steps walk through setting up the patching example policy set:

1. In the Cisco ISE Web Console, click **Policy** > **Policy Sets**.
2. Click the Add icon.
3. Under the **Policy Set Name** field, type: Wired.
4. Click the plus button under the **conditions** field.
5. In the **Editor** field. select **Click to add an attribute field**.
6. Click and drag over the **Wired_802.1X** and **Wired_MAB** conditions from the **library** field.
7. Click the **Use** button.
8. Click the arrow under **View** on the newly created Wired policy.
9. Under the **Authentication Policy** tab, add the policy depicted below. It allows Cisco ISE to authenticate 802.1x users against an identity store. The identity store we used was our AD users.



10. Under the **Authorization Policy** tab, three new rules need to be created, as the screenshot below depicts. The Posture-NonCompliant rule says that devices that are assessed and deemed not compliant should be assigned the Quarantined_Systems security group tag. The Posture rule says that devices that are marked compliant should be permitted access to the network and assigned an employee group tag. The Posture-Unknown rule states that devices that have an unknown posture, meaning the device has yet to be assessed by Cisco ISE, should be redirected to install the posture assessment module.

Rule Name	Conditions	Profiles	Security Groups
Posture-NonCompliant	Non_Compliant_Devices	× NonCompliantAccept	Quarantined_Systems
Posture	Compliant_Devices	× PermitAccess × Tenable_Accept	Employees
Posture-Unknown	Compliance_Unknown_Devices	× posture-redirect	Unknown

1262

1263 5.2.8 Client Provisioning Policy

1264 The Cisco AnyConnect module is used by ISE to perform posture assessments of 802.1X and VPN
1265 connected devices. To ensure that users can be provisioned with the latest version of the AnyConnect
1266 module, the Client Provisioning Policy needs to be set up for Windows and macOS devices. Our build
1267 downloaded the Cisco AnyConnect Module to the machine administrating ISE, from the following Cisco
1268 download page, and uploaded the resource during the creation of the Client Provisioning Policy.

1269 Under the Client Provisioning Policy field, the **Windows** and **MAC OS** fields were edited as shown in
1270 Figure 5-2 to provide access for endpoints to download the AnyConnect Module. For more detailed
1271 information regarding setting up Client Provisioning Resources, please consult the following page.

1272 **Figure 5-2 Examples of Client Provisioning Policies**

Windows	If	Any	and	Windows All	and	Condition(s)	then	AnyConnect Configuration And WinSPWizard 2.5.0.1 And Cisco-ISE-NSP
MAC OS	If	Any	and	Mac OSX	and	Condition(s)	then	CiscoTemporalAgentOSX 4.7.00135 And MacOsXSPWizard 2.1.0.42 And Cisco-ISE-NSP

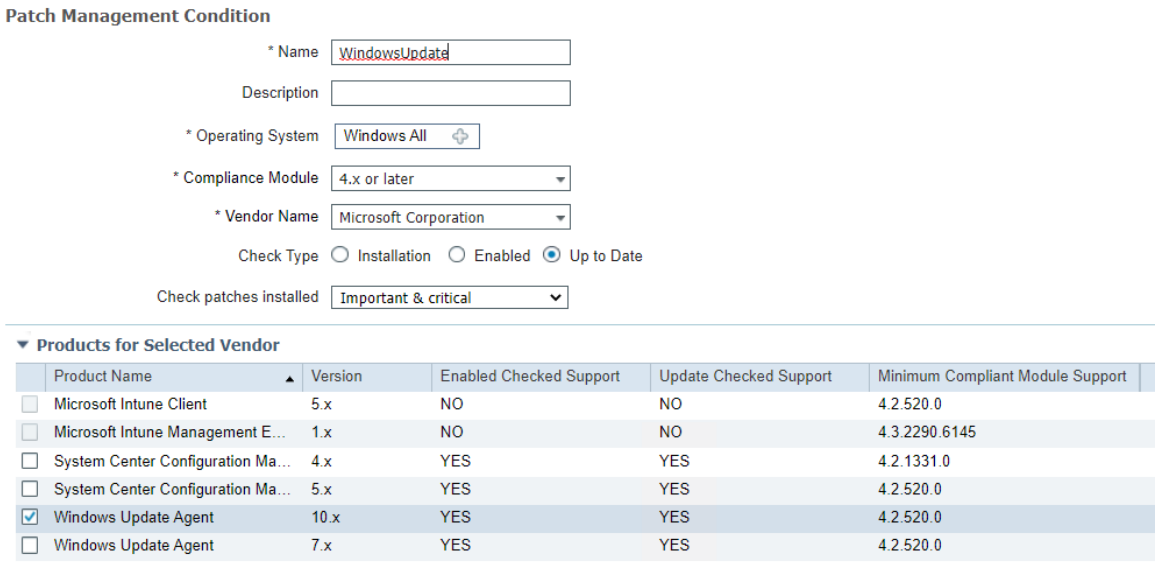
1273 5.2.9 Posture Assessment

1274 The lab instance utilized Cisco ISE's ability to perform posture assessments to determine the patch level
1275 of connected devices. This collected information was used to meet the use case for isolating
1276 unpatchable assets. We configured ISE to perform a posture assessment of a physical Windows laptop.
1277 The posture assessment agent was configured to check if Windows Update reported any missing critical
1278 patches before letting a device join the network. The steps below provide an overview of the work
1279 performed in the lab instance to configure Posture Assessment; more information can be found at the
1280 following page.

1. In the Cisco ISE Web console, click **Policy** > **Posture**.
2. Under the last rule in the list, click the drop-down arrow by the **Edit** button and click **Insert New Policy**.
3. In the policy field, fill out the information as shown below and click **Save**. The policy states that users from any policy group running any version of Windows using the AnyConnect Compliance Module will be subjected to a WinPatching rule that will check for Windows updates.

Policy Options	Update_Windows	If	Any	and	Windows All	and	4.x or later	and	AnyConnect	and	then	WinPatching

4. Click on **Policy** > **Policy Elements** > **Posture** > **Patch Management Condition** to add a new patch management condition. This step configures AnyConnect to check for missing patches for Important and Critical updates on endpoints against Windows Update Agent. Configure the Patch Management Condition with the information shown below.



5. Next, in the ISE interface click **Policy** > **Policy Elements** > **Results** > **Posture** > **Remediation Action**. This step configures ISE to perform remediation actions on devices that are deemed non-compliant. Under the last rule in the list, click the drop-down arrow by the **Edit** button and then click **Insert New Requirement**. Add the following:

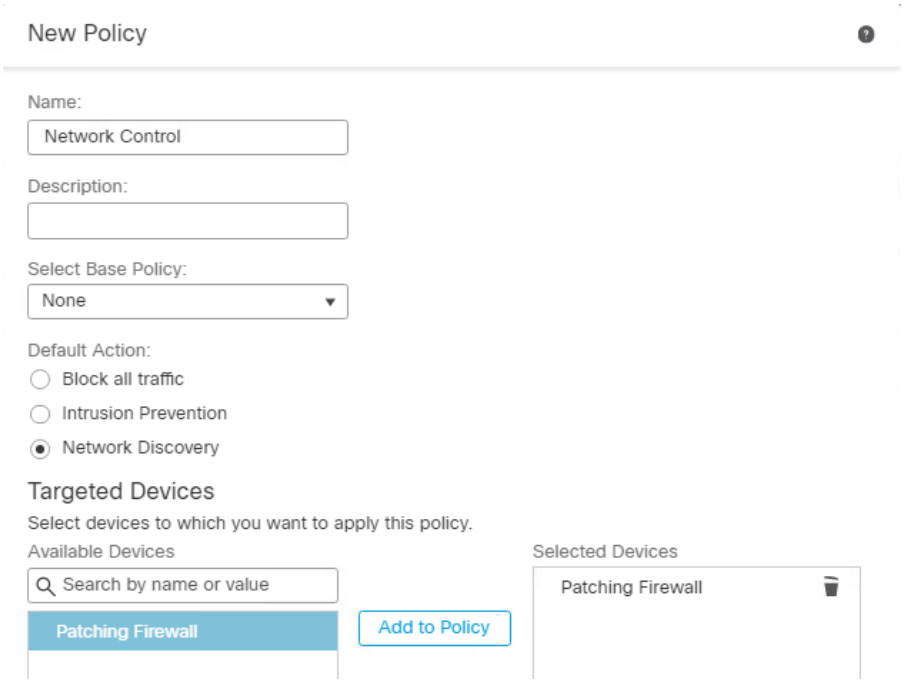
WinPatching	for	Windows All	using	4.x or later	using	AnyConnect	met if	MS	then	WindowsUpdate

6. Click **Save** to save the new requirement.

5.2.10 Cisco FTD Firewall Rules

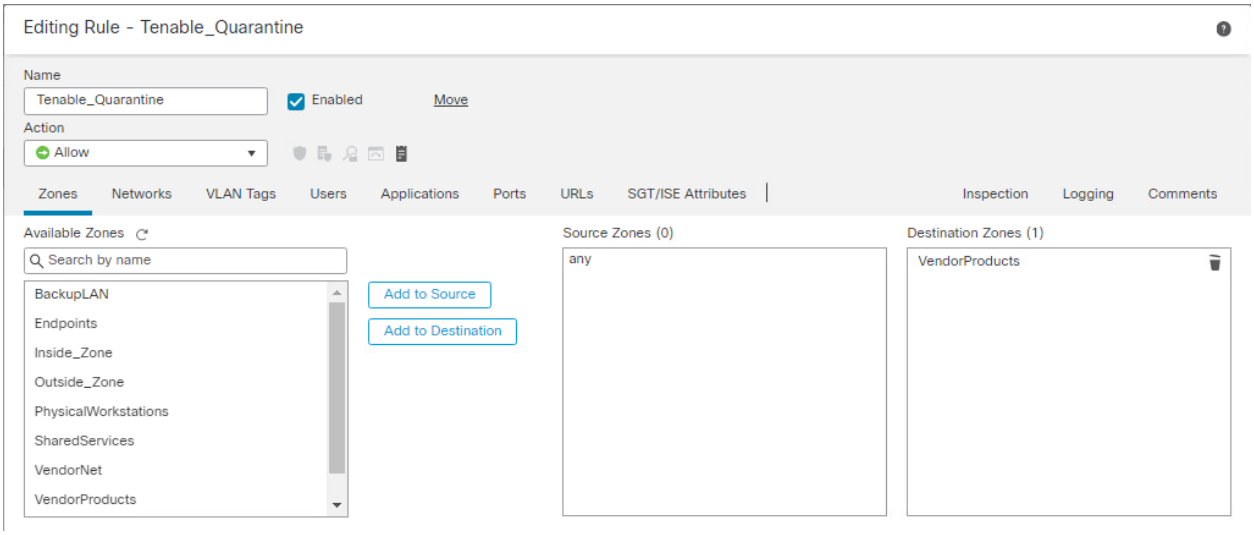
The Cisco FTD firewall rules were used to enforce network restrictions on the quarantined systems using the Quarantined_Systems security group tag in our build. The following steps create a basic enforcement rule:

1. On the Cisco FMC web console, click **Policies** > **Access Control**.
2. Click **New Policy**.
3. Fill out the New Policy Form with the information below. The default action for the policy is network discovery. Network discovery allows for traffic to be monitored by the firewall only without blocking traffic. This monitored traffic is then collected and can be utilized by network admins to create organizational firewall rules. Once firewall rules are in place for your organization, this item can be changed to block all traffic.

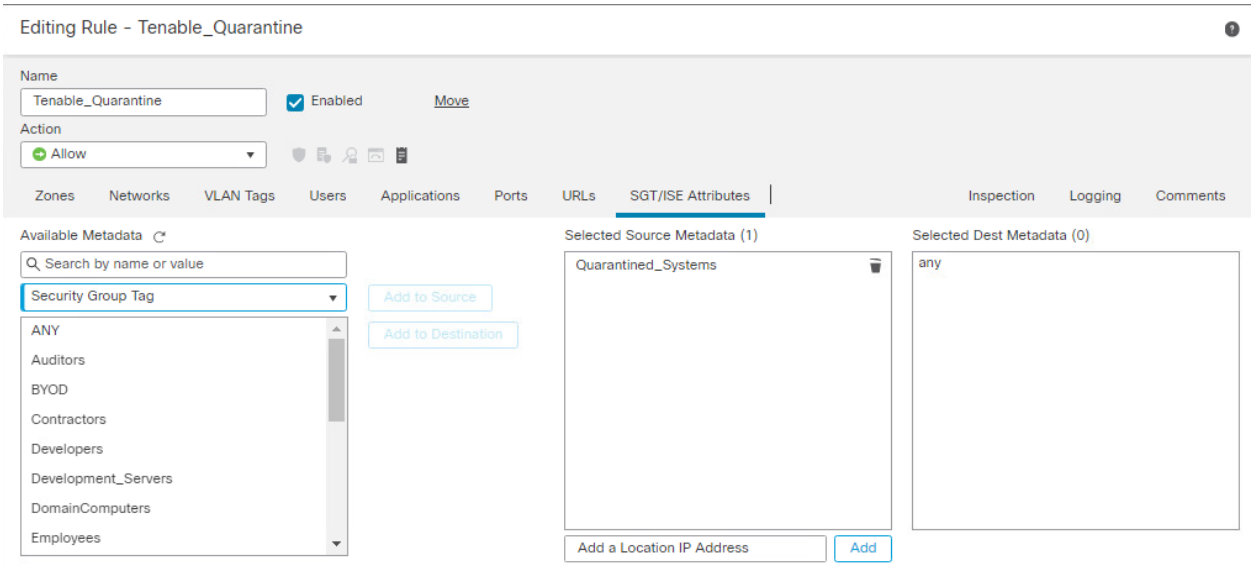


4. Click **Save**.
5. Click the edit button on the newly created rule.
6. Click the **Add Rule** button.
7. Under the **Zones** category, add the information in the screenshot below. The rule states that traffic that is coming from anywhere will be allowed to the VendorProducts zone, which contains the vendor-supplied patching products that were utilized in this build. This rule ensures

that quarantined systems can still receive patches and updates from the appropriate patching system.



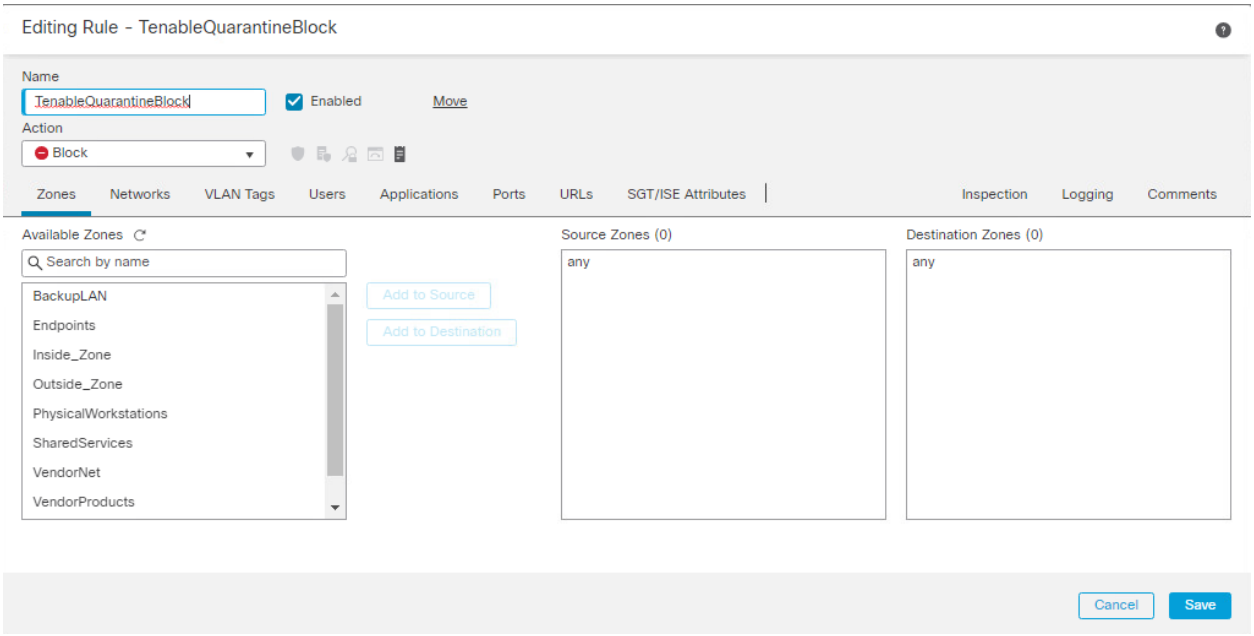
8. Under the **SGT/ISE Attributes** tab, fill out the fields with the information in the screenshot. This applies the network access control from step 7 only to traffic that originates from a machine with the Quarantined_Systems security group tag.



9. Click **Save**.

10. Click the **Add Rule** button to add an additional rule.

11. Edit the **Zones** tab with the information in the screenshot. This rule causes any traffic that has the Quarantined_Systems security group tag to be blocked from traversing the network.



12. Under the **SGT/ISE Attributes** tab, add the Quarantined_Systems security group tag to Selected Source Metadata, like in step 8.

5.3 Cisco Maintenance

All Cisco products should be kept up to date. You are required to have an active Cisco account to download updated software. Software for all Cisco products can be downloaded from here. Follow the guidance on the following pages to upgrade the Cisco product of your choice:

- Upgrade Cisco Firepower Management Center
- Upgrade Cisco Firepower Threat Defense
- Upgrade Cisco Identity Services Engine

6 Microsoft

In this implementation, we used Microsoft Endpoint Configuration Manager to perform configuration management, including software and firmware patching. Microsoft Endpoint Configuration Manager also provided discovery capabilities for endpoints and the capability to respond to emergency scenarios, such as providing a workaround or an emergency patch.

6.1 Microsoft Installation and Configuration

Our implementation utilized a standalone deployment of Microsoft Endpoint Configuration Manager with a separate instance of the database server running Microsoft SQL 2019. The Microsoft Endpoint Configuration Manager was configured to manage multiple Windows-based hosts within the lab environment. The standalone server hosting the Microsoft Endpoint Configuration Manager and the SQL Server were running Windows Server 2019. Each of these servers was joined to the lab Domain Controller, allowing Microsoft Endpoint Configuration Manager to utilize the services the Domain Controller provided. Information on how to determine the correct deployment for your environment can be found here.

Our implementation of Endpoint Configuration Manager consisted of multiple components, including:

- **Windows Server Update Services (WSUS)**, an update service primarily used for downloading, distributing, and managing updates for Microsoft Windows-based systems. Information on how to deploy the WSUS role on Windows Server 2019 can be found here.
- **Microsoft SQL Server**, which served as a database for the Endpoint Configuration Manager sites. The sites are where most of the data for the Endpoint Configuration Manager product is stored. Information on how to deploy Microsoft SQL Server 2019 can be found here.
- **Microsoft Endpoint Configuration Manager site server**, which hosted the core functionality of Endpoint Configuration Manager. Microsoft Endpoint Configuration Manager sites are used to manage endpoints. Information on how to deploy the Endpoint Configuration Manager sites can be found here.
- **Microsoft Endpoint Configuration Manager console**, which was needed to perform administration tasks and was the interface for interacting with the Endpoint Configuration Manager sites. Information on how to deploy the Endpoint Configuration Manager console can be found here.

6.2 Device Discovery

In our implementation, we utilized Heartbeat Discovery, Active Directory System, and Active Directory Group Discovery. Heartbeat Discovery functioned by having the Microsoft Endpoint Configuration Manager agent on the endpoint periodically communicate with the Microsoft Endpoint Configuration Manager server. Active Directory System and Active Directory Group Discovery took advantage of the Enterprise Patching domain and retrieved domain information from the directory server on computers joined to the domain and groups.

More information on how to set up device discovery capabilities can be found here.

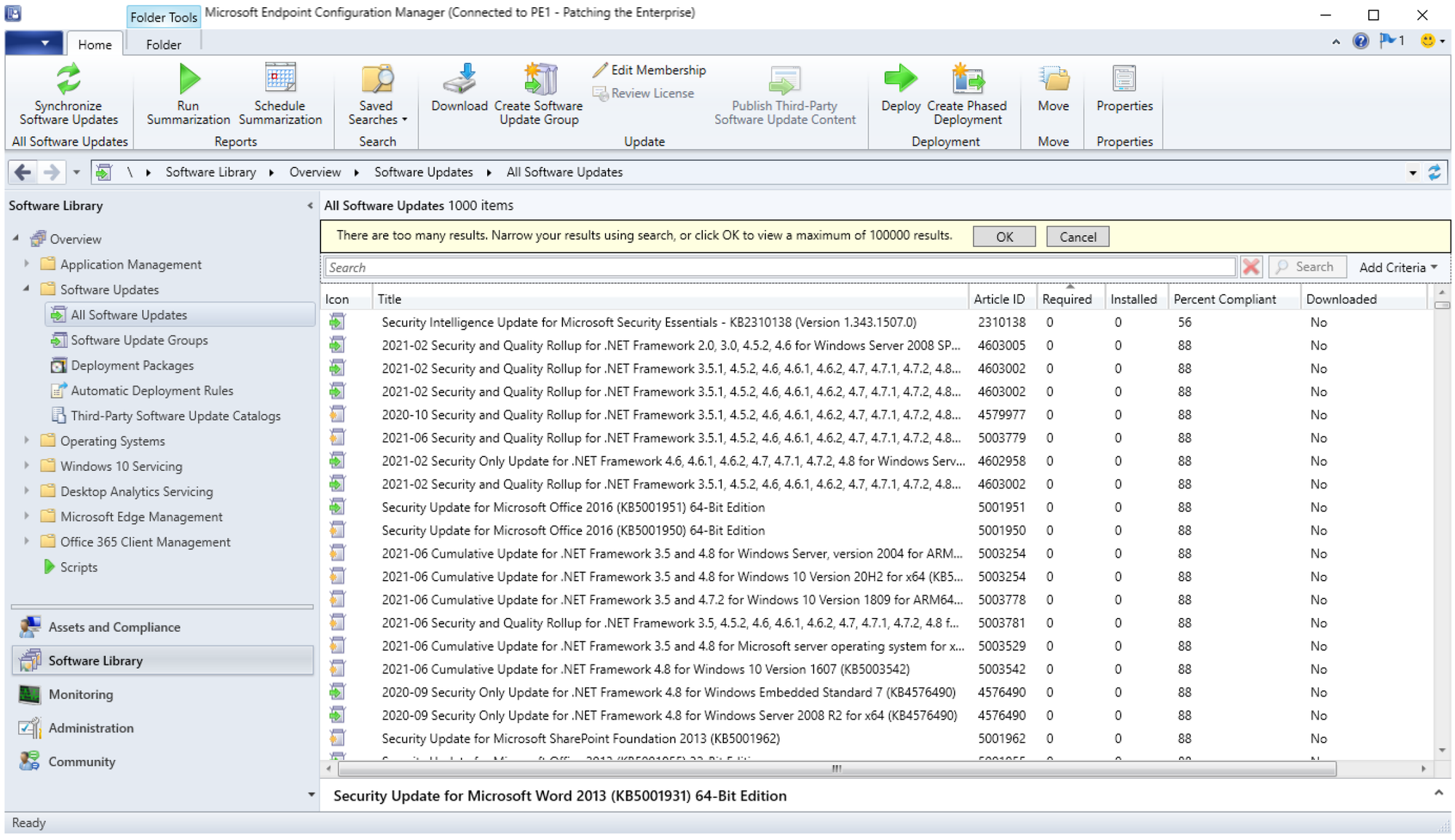
6.3 Patching Endpoints with Microsoft Endpoint Configuration Manager

For our implementation, Microsoft Endpoint Configuration Manager was configured to support software updates to Windows devices. More information on how to do this can be found here.

Our deployment relied on third-party updates to deploy non-Microsoft-based software updates. The implementation subscribed to update catalogues that supported software updates for firmware. More information on how to configure third-party updates can be found here.

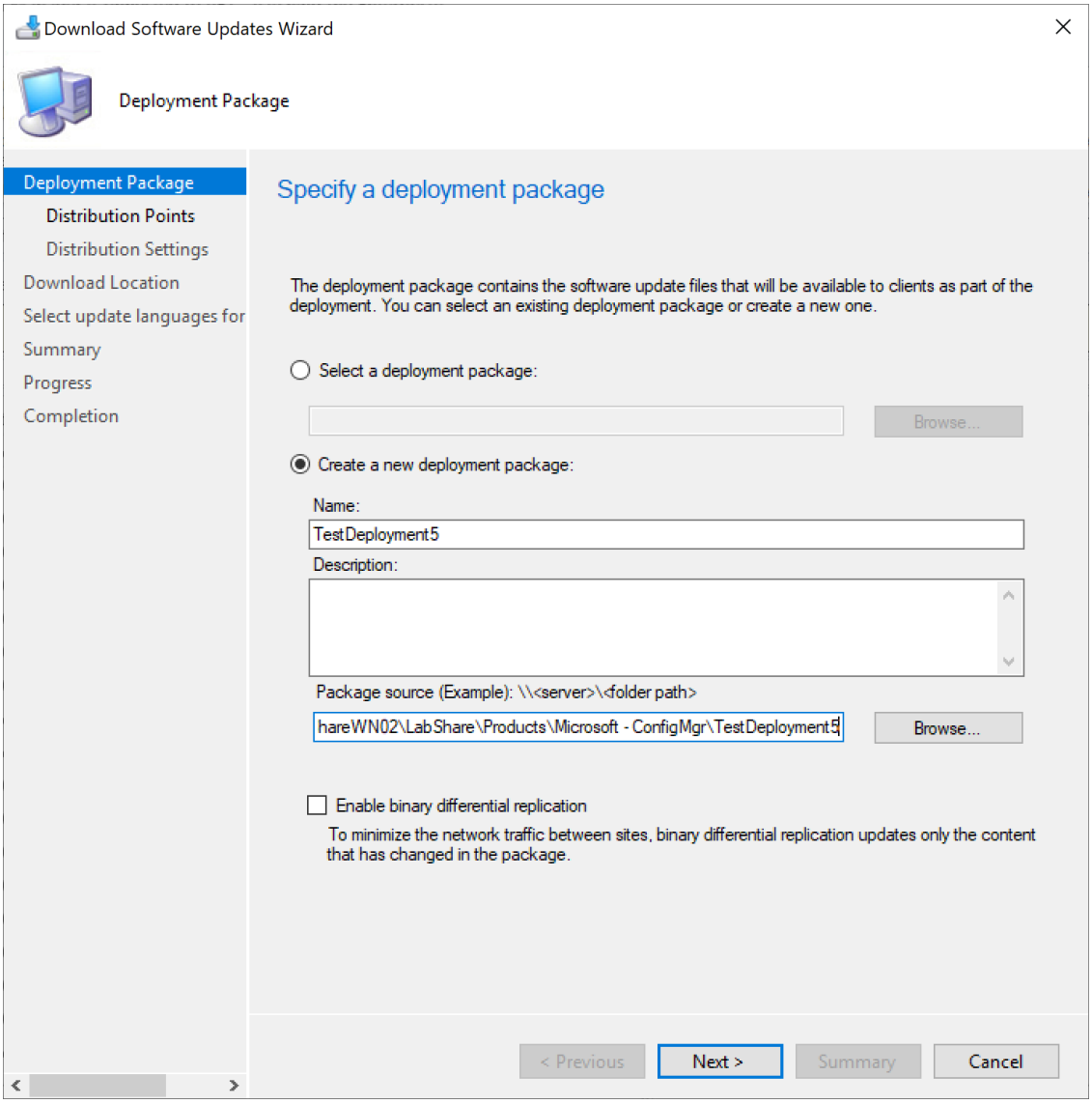
Although there are multiple methods for distributing patches, our deployment utilized the manual method for deploying software updates. This method applied to both third-party updates and updates from Microsoft. This was achieved by first downloading the software updates we wanted to deploy from the "All Software Updates" view, as Figure 6-1 shows. From this view you can download the software updates you want to deploy.

1387 **Figure 6-1 All Software Updates View for Microsoft Endpoint Configuration Manager**



From this view you can download the software updates you want to deploy. The next step we performed was creating a new deployment package. Figure 6-2 provides an example of this.

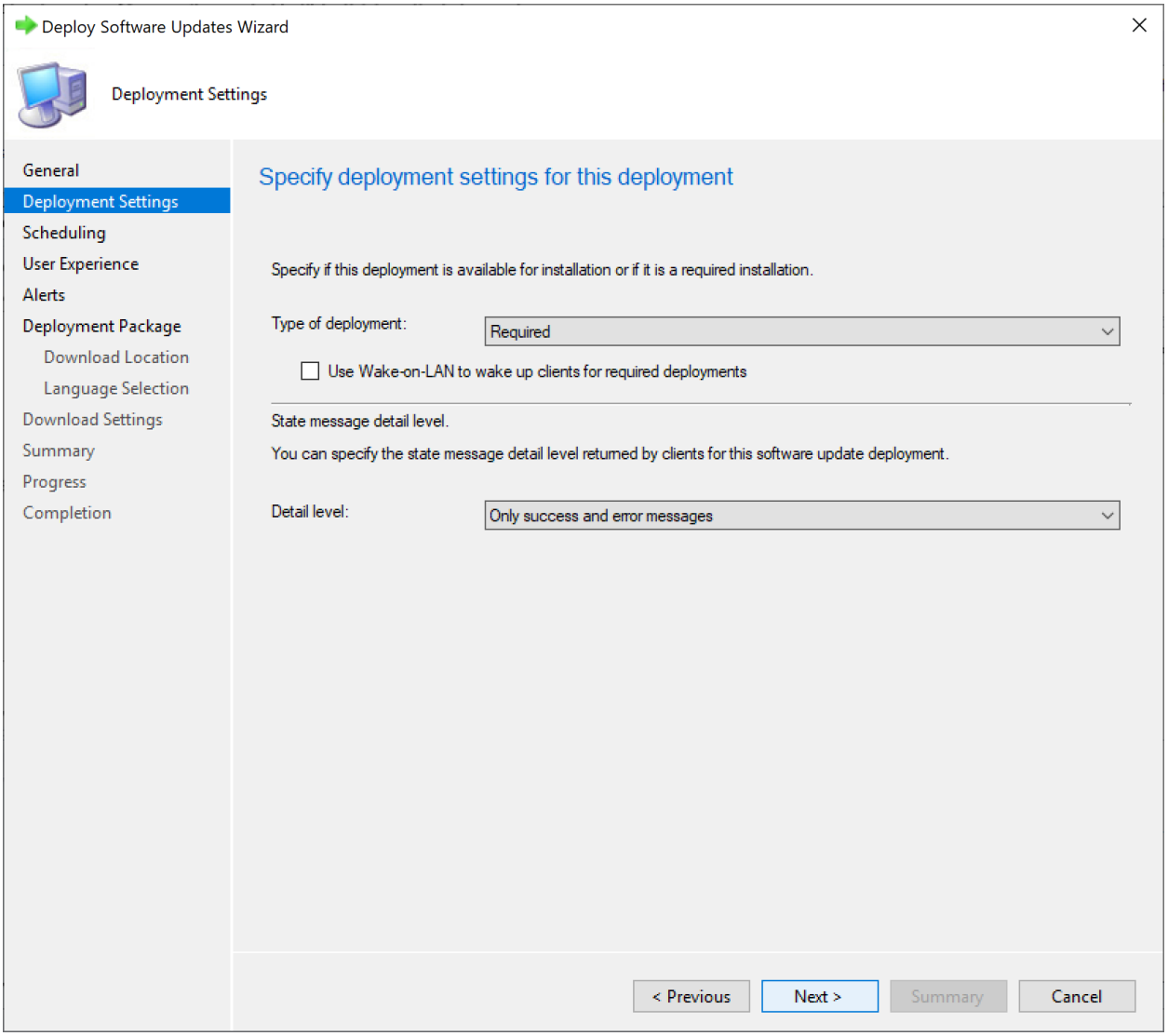
Figure 6-2 Creating a New Deployment Package with Microsoft Endpoint Configuration Manager



After creating a deployment package, the updates can be distributed to endpoints by adding the deployment package to a software update group. More information on how to use this method can be found here.

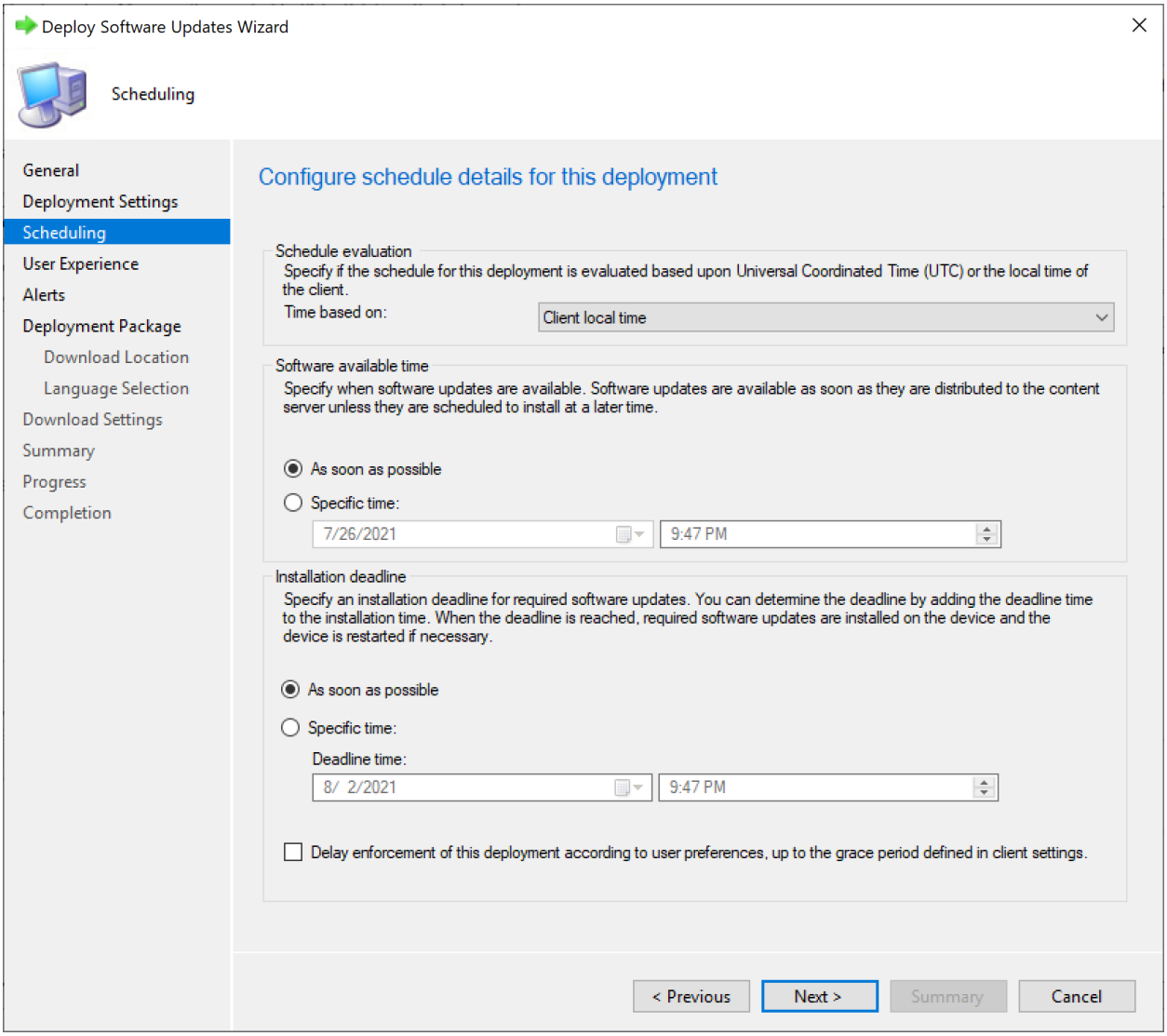
For instances where updates need to be deployed more quickly, deployments can be specified with immediate delivery by changing the deployment type to **Required**. See Figure 6-3 showing the settings for an example deployment.

Figure 6-3 Deployment Settings



This forces the update to be installed based on the schedule specified in the deployment. For immediate updates, select **As soon as possible** when configuring the schedule for deployment. Figure 6-4 shows the schedule details for an example deployment.

1401 **Figure 6-4 Deployment Schedule**



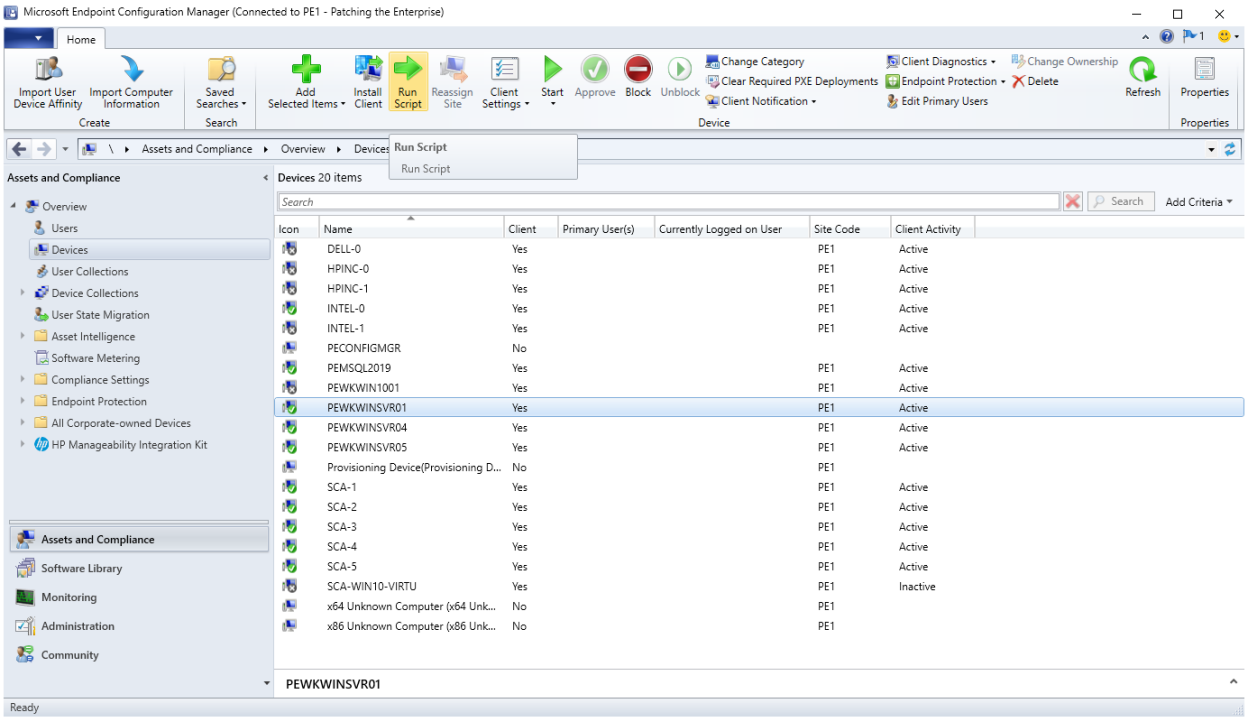
1402 Our deployment also relied on Endpoint Configuration Manager's ability to deploy a PowerShell script to
1403 endpoints for emergency workaround scenarios. We utilized a script that uninstalled Java on the
1404 endpoint on which the script is run. More information on how to deploy PowerShell scripts can be found
1405 here.

1406 The script we uploaded into Microsoft Endpoint Configuration Manager for the build was:

1407
```
gwmi Win32_Product -filter "name like 'Java%'" | % { $_.Uninstall() }
```

Our deployment relied on Microsoft Endpoint Configuration Manager's ability to deploy scripts directly to endpoints. This was achieved by selecting an endpoint from the **Devices** view and selecting the **Run Script** option, as Figure 6-5 illustrates.

Figure 6-5 Devices View with Run Script Option Selected

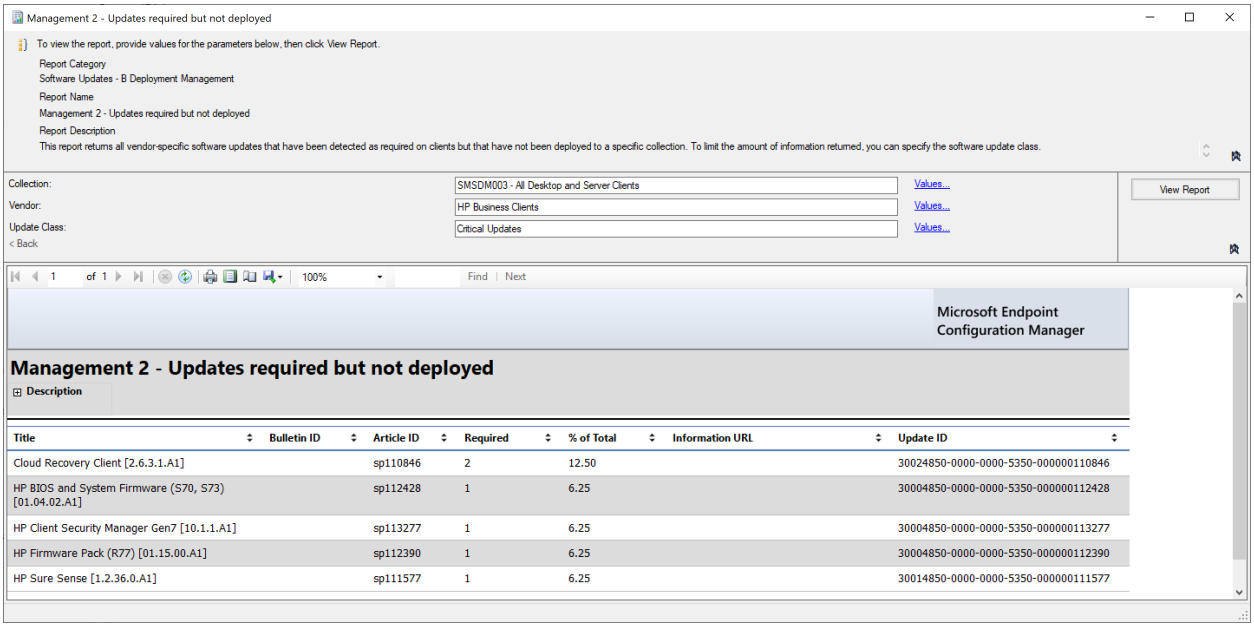


6.4 Microsoft Reporting

We utilized the reporting capabilities of Microsoft Endpoint Configuration Manager to determine which Windows patches and third-party updates were available for endpoints. Information on configuring those reporting capabilities can be found here.

The build utilized the available Software Updates reports from Microsoft Endpoint Configuration Manager to determine specific software updates that were available for endpoints. An example of a report used for this to determine what critical third-party updates are available can be seen in Figure 6-6.

1420 **Figure 6-6 Report Showing Critical 3rd Party Updates Available for HP Business Clients**



1421 6.5 Microsoft Maintenance

1422 Microsoft Endpoint Configuration Manager utilizes in-console updates and servicing. This feature
1423 automatically applies Microsoft-recommended updates that are relevant to your specific infrastructure
1424 and configuration.

1425 7 Forescout

1426 In this implementation, we used the Forescout platform to perform endpoint discovery. The Forescout
1427 platform can perform endpoint discovery by detecting endpoints and determining software information
1428 about those endpoints based on a set of attributes. The Forescout platform also provided the capability
1429 to isolate or restrict unpatchable assets and to respond to emergency scenarios, such as providing a
1430 workaround or deploying an emergency patch. This section explains how the Forescout platform was
1431 utilized in this build.

1432 7.1 Installation and Configuration of Enterprise Manager and Appliance

1433 Our implementation of the Forescout platform utilized both the Forescout Enterprise Manager and
1434 Forescout Appliance. Instructions for deploying these can be found here.

1435 In our setup, the Enterprise Manager allowed for the management of multiple Forescout Appliances.
1436 Although our build only contained one appliance, we chose to utilize the Enterprise Manager to

demonstrate an enterprise environment and to enable adding more appliances to our build if needed. The Forescout Appliance was deployed to have a dedicated virtual device for monitoring network traffic. Depending on the size of your network and your specific requirements, more than one physical or virtual appliance may be recommended.

7.1.1 Installation via OVF

Instructions for deploying OVF templates that can be utilized as either an Enterprise Manager or Forescout Appliance can be found here. The OVF installation method was used by the team for both the Enterprise Manager and Appliance deployment; there are other installation methods available that may be better suited for your environment.

7.1.2 Installation of Forescout Console and Initial Setup

The console application is required to complete the installation of the Forescout platform and to administer the system. The console was installed on a dedicated VM running the Windows 10 OS. This VM has network access to the Forescout Enterprise Manager and Appliance. The instructions for initial installation and setup of the Forescout console can be found here.

7.2 Forescout Capabilities Enabled

After installation and initial setup, it is recommended to enable additional capabilities for the Forescout platform to utilize. The capabilities enabled will depend on what services are available in your environment for Forescout to integrate with. The following subsections cover the basic options the team enabled and utilized in our build.

7.2.1 Network

The Forescout platform was configured to capture network traffic from the Forescout Appliance. Traffic was collected from all of the internal subnets from the lab environment. This allowed the Forescout Appliance to identify hosts on our network by collecting network traffic from the virtual switch using a mirror port. This allowed traffic to be collected from endpoints without requiring an agent or communicating directly between the Forescout platform and the endpoints.

7.2.2 User Directory

The User Directory plugin was configured so that the Forescout platform integrated with the lab's AD Domain Controller. This plugin provided Lightweight Directory Access Protocol (LDAP) services to Forescout, allowing directory-based users to log in into Forescout as well as providing user directory information such as the current active domain users logged into each endpoint. More information about this plugin can be found in the *Authentication Module: User Directory Plugin Server and Guest Management Configuration Guide*.

7.2.3 DNS Query Extension

This configuration setting allowed Forescout to query the DNS server to determine the hostnames of devices identified by Forescout.

7.2.4 Tenable VM

The Tenable VM plugin provided the Forescout platform with vulnerability and scan status information which can be used to create custom policies. This plugin also enabled Forescout to utilize vulnerability management information that Tenable.sc collected from endpoints, and allowed the Forescout platform to determine if scans had been performed on endpoints within the lab. More information about the Critical Vulnerability Quarantine policy which utilizes the data from this policy can be found in Section 7.3.3. Information on how this plugin can be installed and configured for your environment can be found in the *eyeExtend for Tenable Vulnerability Management Configuration guide*.

7.2.5 Microsoft SMS/SCCM

The Microsoft Systems Management Server (SMS)/System Center Configuration Manager (SCCM) module was configured to allow the Forescout platform to integrate with Microsoft Endpoint Configuration Manager. This module allowed for a custom policy to be created that used data from Microsoft Endpoint Configuration Manager. More information about the SCCM Agent Non Compliant Check policy which utilizes the data from this module can be found in Section 7.3.6. In our build, this module was primarily used to determine which hosts were running the Endpoint Configuration Manager agent and therefore communicating with Microsoft Endpoint Configuration Manager. Information on how this module can be installed and configured for your environment can be found in the *Endpoint Module: Microsoft SMS/SCCM Plugin Configuration guide*.

7.2.6 Linux

The Linux plugin was configured to collect information from and manage Linux-based endpoints via two methods: secure shell (SSH) access to the endpoints, and agent-based integration with the Linux endpoint. Both of these methods for collecting data from endpoints were implemented in the lab environment. Information on how this plugin can be installed and configured for your environment can be found in the *Endpoint Module: Linux SCCM Plugin Configuration guide*.

7.2.7 HPS Inspection Engine

The HPS Inspection Engine was configured to collect information from Windows endpoints via two methods. The first method utilized a directory-based integration with the lab's AD Domain Services instance, which collected domain-based information on the Windows endpoint. The second method utilized an agent-based integration called SecureConnector that allowed Forescout to collect and manage Windows endpoints. The agent-based integration was deployed to endpoints by a Windows

Installer (MSI) installer that was manually downloaded from the Enterprise Manager and installed on the endpoint.

Multiple deployment methods can be utilized for installing the SecureConnector. Two methods that were not utilized in this build are automatically deploying software utilizing a configuration management tool, and using a corporate image with the SecureConnector preinstalled when configuring new endpoints for your environment.

Information on how the HPS Inspection Engine can be installed and configured for your environment can be found in the *Endpoint Module: HPS Inspection Engine Configuration guide*.

7.2.8 pxGrid

The pxGrid plugin was configured to integrate with Cisco ISE. This plugin gave the Forescout Platform the ability to utilize Cisco ISE to apply adaptive network control (ANC) policies to endpoints. ANC policies can be used to control network access for endpoints. The ANC policies were enabled on Cisco ISE and could be controlled by third-party systems such as the Forescout platform using pxGrid.

In this implementation, an ANC policy configured within Cisco ISE was used to apply a quarantine policy against the host. For example, in the Critical Vulnerability Quarantine Policy in Section 7.3.3, Forescout communicates to Cisco ISE to quarantine the host when critical vulnerabilities are found on the endpoint via the Tenable VM plugin. After the Cisco ISE ANC policy is applied to a host, the device is assigned a Quarantine security group tag by Cisco ISE. The pxGrid integration between ISE and the Cisco FTD firewall allows for security group tags to be shared. This SGT is then applied by ISE, and network traffic at layer 3 is controlled via firewall rules that were created in Section 5.2.7. Information on how this plugin can be installed and configured for your environment can be found in the *pxGrid Plugin Configuration guide*.

7.2.9 Switch

The Switch plugin was configured to integrate the Forescout platform with the physical Cisco switch located in the lab. The plugin used information from the switch to collect information about endpoints that were physically connected to the switch. Information on how this plugin can be installed and configured for your environment can be found in the *Network Module: Switch Plugin Configuration guide*.

7.2.10 VMWare vSphere/ESXi

Forescout can integrate with VMWare vCenter or ESXi host via a plugin. Our build utilized this plugin to collect information on what virtual hosts and appliances were running in support of the host discovery scenario. We configured Forescout to collect information from a VMWare ESXi host installed on a Dell

R620 server in the lab environment. Information on how this plugin can be installed and configured can be found on the following page.

The following is an overview of the steps for configuring the plugin:

1. Open the Forescout Console and go to **Options** > **Tools**.
2. Select **VMWare vSphere** from the left pane.
3. Select **Add**.
4. Fill out the resulting form with the requested parameters.

7.3 Policies

The project received policies from Forescout that are normally made available to a customer when they purchase professional services from Forescout. These policies helped the team to discover, classify, and assess endpoints on the lab network. More information on how to receive the professional services policies can be found here.

To satisfy the scenarios outlined in the project description, the team also created the following custom policies. More information on how to create custom policies can be found here.

7.3.1 Adobe Flash Player Removal Policy

The Adobe Flash Player Removal policy checks if Flash is running on a Windows Endpoint. If it is, this policy will terminate the process running Flash and uninstall Flash by running the command "`uninstall_flash_player.exe -uninstall`" on the endpoint.

```
<RULES>
<RULE APP_VERSION="8.2.2-
731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DES
CRIPTION=" " ENABLED="true" ID="-2605681954930199910" NAME="Adobe Flash Player
Removal" NOT_COND_UPDATE="true" UPGRADE_PERFORMED="true">
<GROUP_IN_FILTER>
<GROUP ID="1391284960034120761" NAME="Windows"/>
</GROUP_IN_FILTER>
<INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
<ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
<MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
<ADMISSION ALL="true"/>
</MATCH_TIMING>
```

```
<EXPRESSION EXPR_TYPE="AND">
<!-- Rule expression. Rule name is: Adobe Flash Player Removal -->
<EXPRESSION EXPR_TYPE="SIMPLE">
<CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext" LABEL="Windows
Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="HPS Inspection
Engine" PLUGIN_UNIQUE_NAME="va" PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="
111020046" RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
<FILTER CASE_SENSITIVE="false" FILTER_ID="-
8737023325596837863" TYPE="contains">
<VALUE VALUE2="Flash"/>
</FILTER>
</CONDITION>
</EXPRESSION>
<EXPRESSION EXPR_TYPE="SIMPLE">
<CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="nbthost" LABEL="NetBIOS
Hostname" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="NBT
Scanner" PLUGIN_UNIQUE_NAME="nbtscan_plugin" PLUGIN_VESRION="3.2.1" PLUGIN_VESR
ION_NUMBER="32010012" RET_VALUE_ON_UKNOWN="IRRESOLVED" RIGHT_PARENTHESIS="0">
<FILTER CASE_SENSITIVE="false" FILTER_ID="-5759361289894250 39" TYPE="contains">
<VALUE VALUE2="PEWKWINSVR02"/>
</FILTER>
</CONDITION>
</EXPRESSION>
</EXPRESSION>
<EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH">
<RANGE FROM="10.131.5.2" TO="10.131.5.2"/>
<RANGE FROM="10.132.2.11" TO="10.132.2.11"/>
</EXCEPTION>
<EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
<EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
<EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
<EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
<ORIGIN NAME="CUSTOM"/>
```

```
<UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
<ADMISSION ALL="true"/>
</UNMATCH_TIMING>
<SEGMENT ID="2960766429758300381" NAME="Endpoints">
<RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
<RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
<RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
</SEGMENT>
<RULE_CHAIN>
<INNER_RULE APP_VERSION="8.2.2-
731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DES
CRIPTION="" ID="1600971908334654081" NAME="Runing
Flash" NOT_COND_UPDATE="true" RECHECK_MAIN_RULE_DEF="true">
<MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
<ADMISSION ALL="true"/>
</MATCH_TIMING>
<EXPRESSION EXPR_TYPE="SIMPLE">
<!-- Rule expression. Rule name is: Runing Flash -->
<CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext" LABEL="Windows
Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="HPS Inspection
Engine" PLUGIN_UNIQUE_NAME="va" PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="
111020046" RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
<FILTER CASE_SENSITIVE="false" FILTER_ID="2547115646639713943" TYPE="contains">
<VALUE VALUE2="Flash"/>
</FILTER>
</CONDITION>
</EXPRESSION>
<ACTION DISABLED="false" NAME="add-to-group">
<PARAM NAME="temporary" VALUE="true"/>
<PARAM NAME="group-name" VALUE="id:-6458612277141846421;name:Adobe Flash
Running"/>
<PARAM NAME="item_key" VALUE="mac_or_ip"/>
<PARAM NAME="comment" VALUE=""/>
```

```
<SCHEDULE>
<START Class="Immediately"/>
<OCCURENCE onStart="true"/>
</SCHEDULE>
</ACTION>
</INNER_RULE>
<INNER_RULE APP_VERSION="8.2.2-
731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DES
CRIPTION="Upload the uninstaller into the script repository and have it push to
the endpoint and execute it with the silent uninstall option?
https://fpdownload.macromedia.com/get/flashplayer/current/support/uninstall_fla
sh_player.exe uninstall_flash_player.exe -uninstall Or to uninstall a specific
player type (ActiveX, NPAPI, or PPAPI), use the following:
uninstall_flash_player.exe -uninstall activex uninstall_flash_player.exe -
uninstall plugin uninstall_flash_player.exe -uninstall pepperplugin" ID="-
7555287754841043925" NAME="Uninstall Adobe
Flash" NOT_COND_UPDATE="true" RECHECK_MAIN_RULE_DEF="true">
<MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
<ADMISSION ALL="true"/>
</MATCH_TIMING>
<EXPRESSION EXPR_TYPE="SIMPLE">
<!-- Rule expression. Rule name is: Uninstall Adobe Flash -->
<CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext" LABEL="Windows
Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="HPS Inspection
Engine" PLUGIN_UNIQUE_NAME="va" PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="
111020046" RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
<FILTER CASE_SENSITIVE="false" FILTER_ID="2974243046085011295" TYPE="contains">
<VALUE VALUE2="FLASH"/>
</FILTER>
</CONDITION>
</EXPRESSION>
<ACTION DISABLED="true" NAME="process_kill">
<PARAM NAME="process_name" VALUE="flash"/>
<SCHEDULE>
<START Class="Immediately"/>
<OCCURENCE onStart="true"/>
```

```
</SCHEDULE>
</ACTION>
<ACTION DISABLED="true" NAME="run_script">
<PARAM NAME="script_howtorun_ac" VALUE="uninstall_flash_player.exe -
uninstall"/>
<PARAM NAME="script_interactive" VALUE="false"/>
<PARAM NAME="define_time_to_run" VALUE="false"/>
<PARAM NAME="time_to_run" VALUE="1"/>
<SCHEDULE>
<START Class="Immediately"/>
<OCCURENCE onStart="true"/>
</SCHEDULE>
</ACTION>
</INNER_RULE>
</RULE_CHAIN>
<REPORT_TABLES/>
</RULE>
</RULES>
```

7.3.2 Java Removal Policy

The Java Removal policy checks if Java is running on a Windows Endpoint. If it is, this policy will terminate the process running Java and uninstall Java by running a script on the endpoint.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<RULES>
   <RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true"
CLASSIFICATION="REG_STATUS" DESCRIPTION="&#10;&#10; &#10; &#10; "
ENABLED="true" ID="-1659136910494976646" NAME="Java Removal"
NOT_COND_UPDATE="true" UPGRADE_PERFORMED="true">
      <GROUP_IN_FILTER>
         <GROUP ID="1391284960034120761" NAME="Windows"/>
      </GROUP_IN_FILTER>
      <INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
      <ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
```

```
1699            <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
1700               <ADMISSION ALL="true"/>
1701            </MATCH_TIMING>
1702            <EXPRESSION EXPR_TYPE="AND">
1703               <!--Rule expression. Rule name is: Java Removal-->
1704               <EXPRESSION EXPR_TYPE="SIMPLE">
1705                  <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext"
1706        LABEL="Windows Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND"
1707        PLUGIN_NAME="HPS Inspection Engine" PLUGIN_UNIQUE_NAME="va"
1708        PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="111020046"
1709        RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
1710                     <FILTER CASE_SENSITIVE="false" FILTER_ID="3470905276050252920"
1711        TYPE="contains">
1712                        <VALUE VALUE2="Java"/>
1713                     </FILTER>
1714                  </CONDITION>
1715               </EXPRESSION>
1716               <EXPRESSION EXPR_TYPE="SIMPLE">
1717                  <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="nbthost"
1718        LABEL="NetBIOS Hostname" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="NBT
1719        Scanner" PLUGIN_UNIQUE_NAME="nbtscan_plugin" PLUGIN_VESRION="3.2.1"
1720        PLUGIN_VESRION_NUMBER="32010012" RET_VALUE_ON_UKNOWN="IRRESOLVED"
1721        RIGHT_PARENTHESIS="0">
1722                     <FILTER CASE_SENSITIVE="false" FILTER_ID="-575936128989425039"
1723        TYPE="contains">
1724                        <VALUE VALUE2="PEWKWINSVR02"/>
1725                     </FILTER>
1726                  </CONDITION>
1727               </EXPRESSION>
1728            </EXPRESSION>
1729            <EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH">
1730               <RANGE FROM="10.131.5.2" TO="10.131.5.2"/>
1731               <RANGE FROM="10.132.2.11" TO="10.132.2.11"/>
1732            </EXCEPTION>
```

```
            <EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
            <ORIGIN NAME="CUSTOM"/>
            <UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
               <ADMISSION ALL="true"/>
            </UNMATCH_TIMING>
            <SEGMENT ID="2960766429758300381" NAME="Endpoints">
               <RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
               <RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
               <RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
            </SEGMENT>
            <RULE_CHAIN>
               <INNER_RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200"
CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DESCRIPTION="" ID="-
7312022728321489321" NAME="Uninstall Java" NOT_COND_UPDATE="true"
RECHECK_MAIN_RULE_DEF="true">
                  <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                     <ADMISSION ALL="true"/>
                  </MATCH_TIMING>
                  <EXPRESSION EXPR_TYPE="SIMPLE">
                     <!--Rule expression. Rule name is: Uninstall Java-->
                     <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext"
LABEL="Windows Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND"
PLUGIN_NAME="HPS Inspection Engine" PLUGIN_UNIQUE_NAME="va"
PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="111020046"
RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
                        <FILTER CASE_SENSITIVE="false"
FILTER_ID="8761976385823184780" TYPE="contains">
                           <VALUE VALUE2="java"/>
                        </FILTER>
                     </CONDITION>
                  </EXPRESSION>
```

```
                    <ACTION DISABLED="true" NAME="process_kill">
                        <PARAM NAME="process_name" VALUE="java"/>
                        <SCHEDULE>
                            <START Class="Immediately"/>
                            <OCCURENCE onStart="true"/>
                        </SCHEDULE>
                    </ACTION>
                    <ACTION DISABLED="false" NAME="run_script">
                        <PARAM NAME="script_howtorun_ac" VALUE="uninstall_java.ps1"/>
                        <PARAM NAME="script_interactive" VALUE="false"/>
                        <PARAM NAME="define_time_to_run" VALUE="false"/>
                        <PARAM NAME="time_to_run" VALUE="10"/>
                        <SCHEDULE>
                            <START Class="Immediately"/>
                            <OCCURENCE onStart="true"/>
                        </SCHEDULE>
                    </ACTION>
                </INNER_RULE>
                <INNER_RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200"
CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DESCRIPTION="" ID="-
8890693029182562272" NAME="Runing Java" NOT_COND_UPDATE="true"
RECHECK_MAIN_RULE_DEF="true">
                    <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                        <ADMISSION ALL="true"/>
                    </MATCH_TIMING>
                    <EXPRESSION EXPR_TYPE="SIMPLE">
                        <!--Rule expression. Rule name is: Runing Java-->
                        <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="process_no_ext"
LABEL="Windows Processes Running" LEFT_PARENTHESIS="0" LOGIC="AND"
PLUGIN_NAME="HPS Inspection Engine" PLUGIN_UNIQUE_NAME="va"
PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="111020046"
RET_VALUE_ON_UKNOWN="UNMATCH" RIGHT_PARENTHESIS="0">
                            <FILTER CASE_SENSITIVE="false"
FILTER_ID="3138188613733535094" TYPE="contains">
```

```
                                <VALUE VALUE2="Java"/>
                            </FILTER>
                        </CONDITION>
                    </EXPRESSION>
                    <ACTION DISABLED="false" NAME="add-to-group">
                        <PARAM NAME="temporary" VALUE="true"/>
                        <PARAM NAME="group-name" VALUE="id:-
3761570262828389651;name:Java Running"/>
                        <PARAM NAME="item_key" VALUE="mac_or_ip"/>
                        <PARAM NAME="comment" VALUE=""/>
                        <SCHEDULE>
                            <START Class="Immediately"/>
                            <OCCURENCE onStart="true"/>
                        </SCHEDULE>
                    </ACTION>
                </INNER_RULE>
            </RULE_CHAIN>
            <REPORT_TABLES/>
        </RULE>
    </RULES>
```

7.3.3 Critical Vulnerability Quarantine Policy

The Critical Vulnerability Quarantine policy utilizes the Tenable VM plugin to determine if an endpoint has any known critical vulnerabilities. If it does, this policy uses Cisco ISE to quarantine the endpoint by utilizing the pxGrid plugin.

```
    <?xml version="1.0" encoding="UTF-8" standalone="no"?>
    <RULES>
        <RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true"
CLASSIFICATION="REG_STATUS" DESCRIPTION="" ENABLED="true" ID="-
6639485913457214440" META_TYPE="COMPLY" NAME="Forescout Critical Vulnerability
Quarantine" NOT_COND_UPDATE="true" UPGRADE_PERFORMED="true">
            <GROUP_IN_FILTER/>
            <INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
```

```
1833            <ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
1834            <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
1835               <ADMISSION ALL="true"/>
1836            </MATCH_TIMING>
1837            <EXPRESSION EXPR_TYPE="SIMPLE">
1838               <!--Rule expression. Rule name is: Forescout Critical Vulnerability
1839         Quarantine-->
1840               <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="nbthost"
1841         LABEL="NetBIOS Hostname" LEFT_PARENTHESIS="0" LOGIC="AND" PLUGIN_NAME="NBT
1842         Scanner" PLUGIN_UNIQUE_NAME="nbtscan_plugin" PLUGIN_VESRION="3.2.1"
1843         PLUGIN_VESRION_NUMBER="32010012" RET_VALUE_ON_UKNOWN="IRRESOLVED"
1844         RIGHT_PARENTHESIS="0">
1845                  <FILTER CASE_SENSITIVE="false" FILTER_ID="-847734611131793936"
1846         TYPE="contains">
1847                     <VALUE VALUE2="PEWKWINSVR02"/>
1848                  </FILTER>
1849               </CONDITION>
1850            </EXPRESSION>
1851            <EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH"/>
1852            <EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
1853            <EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
1854            <EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
1855            <EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
1856            <ORIGIN NAME="CUSTOM"/>
1857            <UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
1858               <ADMISSION ALL="true"/>
1859            </UNMATCH_TIMING>
1860            <SEGMENT ID="2960766429758300381" NAME="Endpoints">
1861               <RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
1862               <RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
1863               <RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
1864            </SEGMENT>
1865            <RULE_CHAIN>
```

```
            <INNER_RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200"
CACHE_TTL_SYNCED="true" CLASSIFICATION="REG_STATUS" DESCRIPTION="" ID="-
7308160423478365115" NAME="CriticalVuln  pxGrid Policy" NOT_COND_UPDATE="true"
RECHECK_MAIN_RULE_DEF="true">
                <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                    <ADMISSION ALL="true"/>
                </MATCH_TIMING>
                <META_TYPE STATE="NA"/>
                <ACTION DISABLED="false" NAME="apply_anc_policy">
                    <PARAM NAME="policy_name" VALUE="Forescout"/>
                    <SCHEDULE>
                        <START Class="Immediately"/>
                        <OCCURENCE onStart="true"/>
                    </SCHEDULE>
                </ACTION>
                <ACTION DISABLED="false" NAME="balloon_message">
                    <PARAM NAME="msg" VALUE="You have been quarantined. Please
update your computer or contact the helpdesk for assistence."/>
                    <PARAM NAME="look" VALUE="info"/>
                    <SCHEDULE>
                        <START Class="Immediately"/>
                        <OCCURENCE onStart="true"/>
                    </SCHEDULE>
                </ACTION>
            </INNER_RULE>
        </RULE_CHAIN>
        <REPORT_TABLES/>
    </RULE>
</RULES>
```

7.3.4 Force Windows Update Policy

The Force Windows Update policy will force a Windows update on an endpoint with Windows Update enabled by utilizing Forescout's capability to determine if vulnerabilities exist on that endpoint.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<RULES>
   <RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true"
CLASSIFICATION="REG_STATUS" DESCRIPTION="&#10;&#10; &#10; &#10; "
ENABLED="true" ID="8956849743087666010" NAME="Force Windows Update"
NOT_COND_UPDATE="true" UPGRADE_PERFORMED="true">
      <GROUP_IN_FILTER/>
      <INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
      <ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
      <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
         <ADMISSION ALL="true"/>
      </MATCH_TIMING>
      <EXPRESSION EXPR_TYPE="SIMPLE">
         <!--Rule expression. Rule name is: Force Windows Update-->
         <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="vulns"
LABEL="Microsoft Vulnerabilities" LEFT_PARENTHESIS="0" LOGIC="AND"
PLUGIN_NAME="HPS Inspection Engine" PLUGIN_UNIQUE_NAME="va"
PLUGIN_VESRION="11.1.2" PLUGIN_VESRION_NUMBER="111020046"
RET_VALUE_ON_UKNOWN="IRRESOLVED" RIGHT_PARENTHESIS="0">
            <FILTER AUTO_UPDATE="true" FILTER_ID="-32838886002658939"
OPTIONS_DIGEST="b3eaa0cf6df1fc550859e51703f2665a">
               <OPT VALUE="KB890830-141"/>
               <OPT VALUE="KB890830-139"/>
               <OPT VALUE="KB890830-144"/>
               <OPT VALUE="KB890830-138"/>
               <OPT VALUE="KB890830-143"/>
               <OPT VALUE="KB890830-140"/>
               <OPT VALUE="KB890830-148"/>
               <OPT VALUE="KB890830-145"/>
               <OPT VALUE="KB890830-136"/>
               <OPT VALUE="KB890830-151"/>
               <OPT VALUE="KB890830-146"/>
               <OPT VALUE="KB890830-142"/>
               <OPT VALUE="KB890830-147"/>
```

```
                        <OPT VALUE="KB890830-31"/>
                        <OPT VALUE="KB890830-137"/>
                        <OPT VALUE="KB890830-150"/>
                        <OPT VALUE="KB890830-149"/>
                    </FILTER>
                </CONDITION>
            </EXPRESSION>
            <ACTION DISABLED="false" NAME="remediate_wua">
                <PARAM NAME="update_type" VALUE="keep_update_settings"/>
                <PARAM NAME="wsus_target_group" VALUE=""/>
                <PARAM NAME="automatic_updates_type" VALUE="keep_update_settings"/>
                <PARAM NAME="use_default_if_disabled" VALUE="false"/>
                <SCHEDULE>
                    <START Class="Immediately"/>
                    <OCCURENCE onStart="true"/>
                </SCHEDULE>
            </ACTION>
            <EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH">
                <RANGE FROM="10.131.5.2" TO="10.131.5.2"/>
                <RANGE FROM="10.132.2.11" TO="10.132.2.11"/>
            </EXCEPTION>
            <EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
            <EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
            <ORIGIN NAME="CUSTOM"/>
            <UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                <ADMISSION ALL="true"/>
            </UNMATCH_TIMING>
            <SEGMENT ID="2960766429758300381" NAME="Endpoints">
```

```
                <RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
                <RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
                <RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
            </SEGMENT>
            <RULE_CHAIN/>
            <REPORT_TABLES/>
        </RULE>
    </RULES>
```

7.3.5 Agent Compliance Check Policy

The Agent Compliance Check policy will determine if a Windows endpoint has the Microsoft Endpoint Configuration Manager agent installed by seeing if the endpoint has checked in with Endpoint Configuration Manager.

```
    <?xml version="1.0" encoding="UTF-8" standalone="no"?>
    <RULES>
        <RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true"
    CLASSIFICATION="REG_STATUS" DESCRIPTION="" ENABLED="true" ID="-
    3295238297289158 79" NAME="SCCM Agent Compliance" NOT_COND_UPDATE="true"
    UPGRADE_PERFORMED="true">
            <GROUP_IN_FILTER>
                <GROUP ID="1391284960034120761" NAME="Windows"/>
            </GROUP_IN_FILTER>
            <INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
            <ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
            <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                <ADMISSION ALL="true"/>
            </MATCH_TIMING>
            <EXPRESSION EXPR_TYPE="SIMPLE">
                <!--Rule expression. Rule name is: SCCM Agent Compliance-->
                <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="Client_registered"
    LABEL="SMS/SCCM  Client Registration Status" LEFT_PARENTHESIS="0" LOGIC="AND"
    PLUGIN_NAME="Microsoft SMS/SCCM" PLUGIN_UNIQUE_NAME="sms"
    PLUGIN_VESRION="2.4.4" PLUGIN_VESRION_NUMBER="24040014"
    RET_VALUE_ON_UKNOWN="IRRESOLVED" RIGHT_PARENTHESIS="0">
```

```
                <FILTER AUTO_UPDATE="false" FILTER_ID="8830579494271797354"
OPTIONS_DIGEST="93e42278ee53b84f8427494bd2a235c6">
                   <OPT VALUE="db_found_true"/>
                </FILTER>
             </CONDITION>
          </EXPRESSION>
          <ACTION DISABLED="false" NAME="add-to-group">
             <PARAM NAME="temporary" VALUE="true"/>
             <PARAM NAME="group-name" VALUE="id:8255406739413382154;name:SCCM
Client Registered"/>
             <PARAM NAME="item_key" VALUE="mac_or_ip"/>
             <PARAM NAME="comment" VALUE=""/>
             <SCHEDULE>
                <START Class="Immediately"/>
                <OCCURENCE onStart="true"/>
             </SCHEDULE>
          </ACTION>
          <EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH"/>
          <EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
          <EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
          <EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
          <EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
          <ORIGIN NAME="CUSTOM"/>
          <UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
             <ADMISSION ALL="true"/>
          </UNMATCH_TIMING>
          <SEGMENT ID="2960766429758300381" NAME="Endpoints">
             <RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
             <RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
             <RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
          </SEGMENT>
```

```
            <RULE_CHAIN/>
            <REPORT_TABLES/>
         </RULE>
      </RULES>
```

7.3.6 SCCM Agent Non Compliant Check Policy

The SCCM Agent Non Compliant Check policy will determine if a Windows endpoint is non-compliant by seeing if the endpoint has or has not checked into Microsoft Endpoint Configuration Manager.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<RULES>
   <RULE APP_VERSION="8.2.2-731" CACHE_TTL="259200" CACHE_TTL_SYNCED="true"
CLASSIFICATION="REG_STATUS" DESCRIPTION="" ENABLED="true"
ID="6927087801731630440" NAME="SCCM Agent Non Compliant Check"
NOT_COND_UPDATE="true" UPGRADE_PERFORMED="true">
      <GROUP_IN_FILTER/>
      <INACTIVITY_TTL TTL="0" USE_DEFAULT="true"/>
      <ADMISSION_RESOLVE_DELAY TTL="0" USE_DEFAULT="true"/>
      <MATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
         <ADMISSION ALL="true"/>
      </MATCH_TIMING>
      <EXPRESSION EXPR_TYPE="SIMPLE">
         <!--Rule expression. Rule name is: SCCM Agent Non Compliant Check-->
         <CONDITION EMPTY_LIST_VALUE="false" FIELD_NAME="Client_registered"
LABEL="SMS/SCCM  Client Registration Status" LEFT_PARENTHESIS="0" LOGIC="AND"
PLUGIN_NAME="Microsoft SMS/SCCM" PLUGIN_UNIQUE_NAME="sms"
PLUGIN_VESRION="2.4.4" PLUGIN_VESRION_NUMBER="24040014"
RET_VALUE_ON_UKNOWN="IRRESOLVED" RIGHT_PARENTHESIS="0">
            <FILTER AUTO_UPDATE="false" FILTER_ID="-9113011034532548035"
OPTIONS_DIGEST="93e42278ee53b84f8427494bd2a235c6">
               <OPT VALUE="db_found_false"/>
            </FILTER>
         </CONDITION>
      </EXPRESSION>
      <ACTION DISABLED="false" NAME="add-to-group">
```

```
                <PARAM NAME="temporary" VALUE="true"/>
                <PARAM NAME="group-name" VALUE="id:6514702438169432101;name:SCCM
        Missing Agent"/>
                <PARAM NAME="item_key" VALUE="mac_or_ip"/>
                <PARAM NAME="comment" VALUE=""/>
                <SCHEDULE>
                   <START Class="Immediately"/>
                   <OCCURENCE onStart="true"/>
                </SCHEDULE>
             </ACTION>
             <EXCEPTION NAME="ip" UNKNOWN_EVAL="UNMATCH"/>
             <EXCEPTION NAME="mac" UNKNOWN_EVAL="UNMATCH"/>
             <EXCEPTION NAME="nbthost" UNKNOWN_EVAL="UNMATCH"/>
             <EXCEPTION NAME="user" UNKNOWN_EVAL="UNMATCH"/>
             <EXCEPTION NAME="group" UNKNOWN_EVAL="UNMATCH"/>
             <ORIGIN NAME="CUSTOM"/>
             <UNMATCH_TIMING RATE="28800" SKIP_INACTIVE="true">
                <ADMISSION ALL="true"/>
             </UNMATCH_TIMING>
             <SEGMENT ID="2960766429758300381" NAME="Endpoints">
                <RANGE FROM="10.132.6.0" TO="10.132.6.255"/>
                <RANGE FROM="10.151.40.0" TO="10.151.40.255"/>
                <RANGE FROM="192.168.1.0" TO="192.168.1.255"/>
             </SEGMENT>
             <RULE_CHAIN/>
             <REPORT_TABLES/>
          </RULE>
       </RULES>
```

7.4 Forescout Maintenance

Forescout releases suggested updates and plugins in the Forescout Console and through its ActiveCare Maintenance and Support policy.

8 IBM

We used two cloud-based IBM offerings for this build. One, IBM MaaS360 with Watson, was used for endpoint management for desktop and laptop computers in the first phase, and for Android and iOS mobile devices in the second phase. The second offering, the IBM Code Risk Analyzer, was used during the second phase to scan source code in cloud-based containers for vulnerabilities. This section shows how each cloud-based service was configured and used for the build.

8.1 IBM Code Risk Analyzer

The IBM Code Risk Analyzer, a feature of IBM Cloud Continuous Delivery for DevSecOps application architectures, enables developers to quickly assess and remediate security and legal risks that they are potentially introducing into source code and provides them direct actionable feedback. It works with code repositories such as Git to analyze your application, perform a set of compliance control checks, produce a bill of materials, and report vulnerability findings. Code Risk Analyzer is provided as a set of Tekton tasks, which can be easily incorporated into delivery pipelines. Also, it is available as a managed service on IBM Cloud, which eliminates the need to host your own infrastructure to run these delivery pipelines. This section illustrates how we configured Code Risk Analyzer on IBM Cloud to embed and use in development workflows.

8.1.1 Getting Ready

No software installation is required to use Code Risk Analyzer on IBM Cloud. However, make sure you have an active IBM Cloud account.

All the Tekton pipeline definitions for Code Risk Analyzer are open-sourced and publicly available.

We used this sample cloud native micro-service application to demonstrate the configuration and analysis via a delivery pipeline. You need to fork this application under your authorized account for the code repository. If you have any other micro-service application, you can use that as well. Make sure you have WRITE access to the code repository that you plan to use.

8.1.2 Creating Your Toolchain

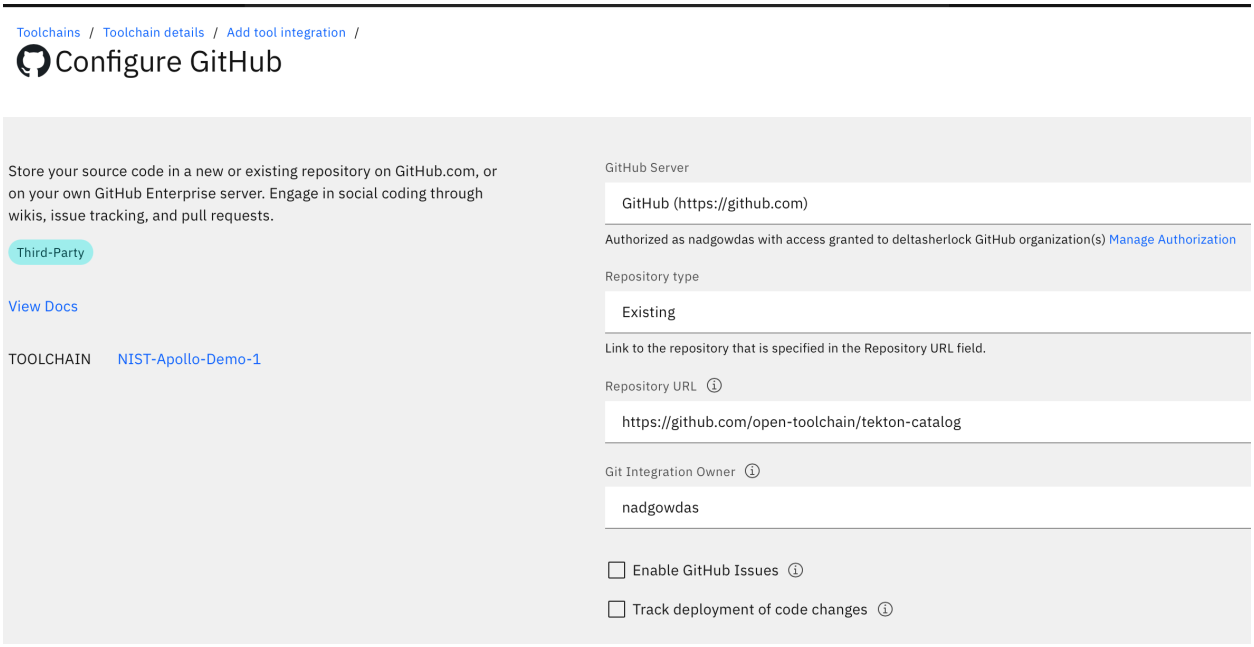
Follow these steps to create and populate your toolchain:

1. Login to your IBM Cloud account and from the service catalog on the left, select **DevOps**. This will open the dashboard for Toolchains.

2. Select **Create toolchain** > **Build your own toolchain**. Give a name to your toolchain and click **Create**.

3. Once the toolchain is created, it needs to be populated with developer tools. Click the **Add tool** button to add the following tools to the toolchain:

    a. Github (code repository for Code Risk Analyzer Tekton definition): configure it as shown below. Note: For a first-time user, it will ask you to authorize IBM Cloud to access your code repository account. This one-time authorization is necessary.

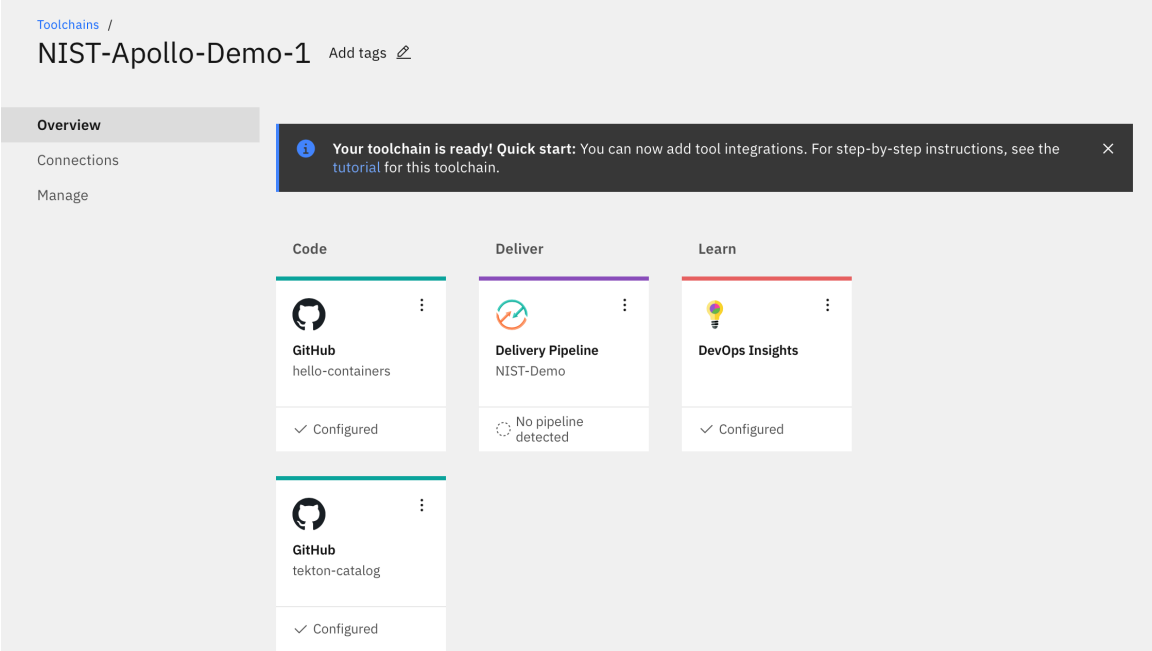


    b. Github (code repository for our sample application): perform a similar integration as above for your application repository.

    c. DevOps Insights (required for authorization and integration): no configuration is necessary, but make sure it is added to your Toolchain workspace.

    d. Delivery Pipeline (automation engine for our pipeline): first give it a **Name** and select "Tekton" as the **Pipeline Type**. The next section contains more detailed information on pipeline configuration.

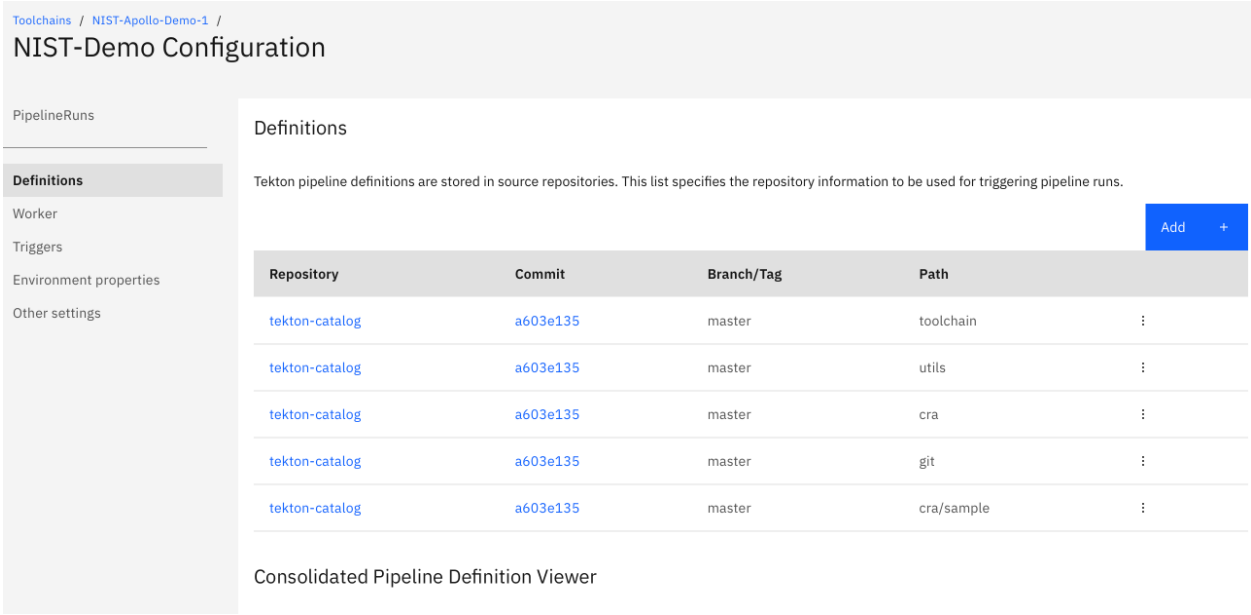
4. At this point, your toolchain workspace should have the tools depicted below.



8.1.3 Configuring Delivery Pipeline

The core logic of configuring Code Risk Analyzer is in the Delivery Pipeline. We need to perform four sections of configuration:

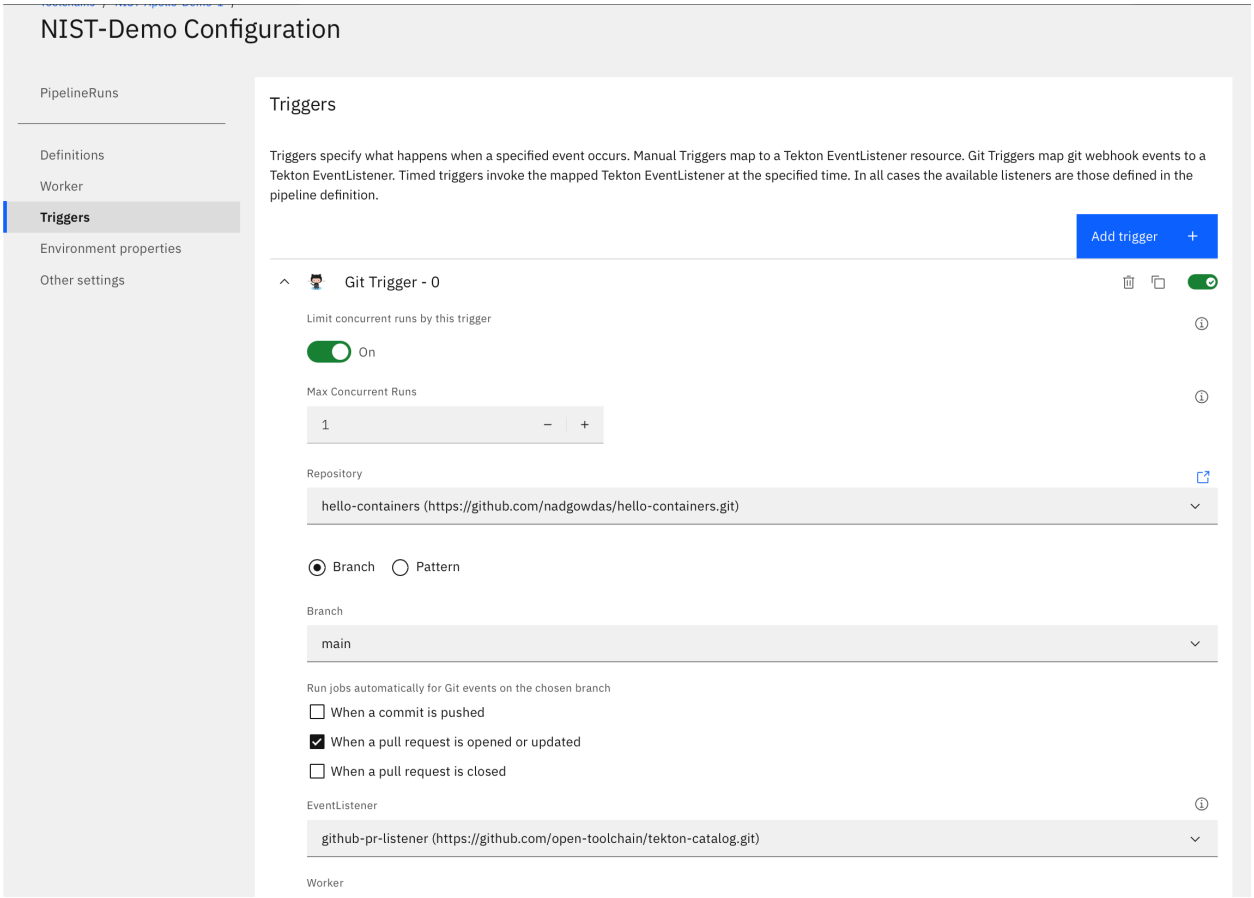
1. **Definition** is where we specify the source for our Code Risk Analyzer pipeline definitions. To do so, click **Add** multiple times to add the following list of locations for sources. When done, click **Save**.



Consolidated Pipeline Definition Viewer

2. **Worker** allows us to select the cluster where the pipeline should execute. We used a managed Kubernetes worker on IBM Cloud to run our pipeline. To do so, select **IBM Managed workers** from the dropdown list.

3. **Trigger** allows us to specify “when” or on which events we want to execute a Code Risk Analyzer scan on a code repository. To configure this, select our sample application code repository, then enable the option to run **when a pull request is opened or updated**.



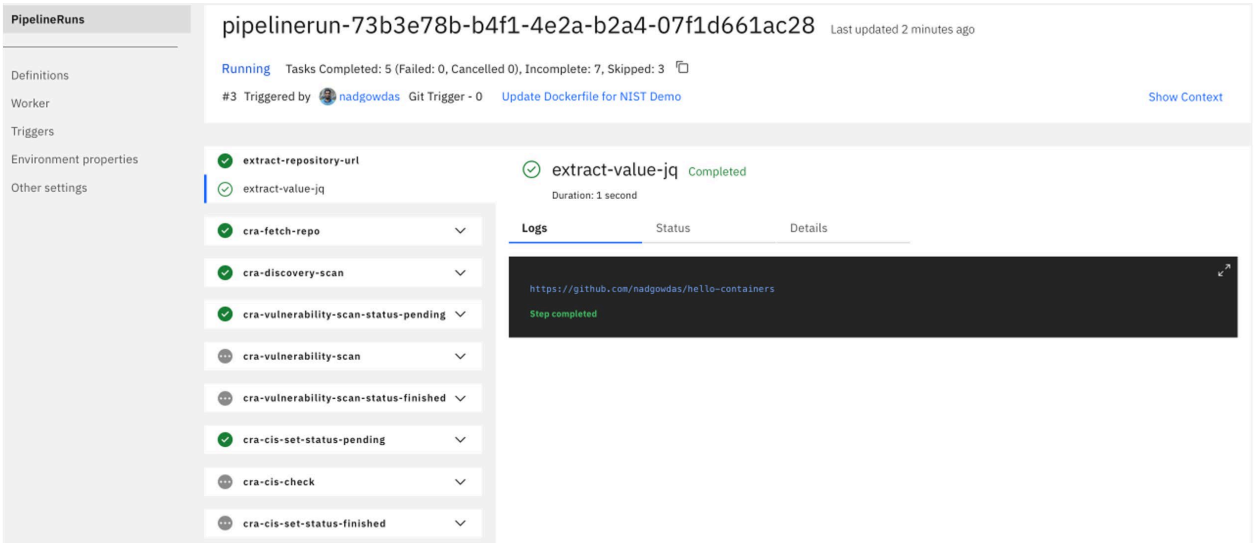
4. **Environment Properties** allows you to store name-value pairs for use in your pipeline. For this build, specify a **Secure value** type named **apikey** with the API_KEY for your IBM Cloud account. You can create a new API_KEY with **Manage** > **Access (IAM)** -> **API Keys**.

At this point, your pipeline is successfully created and configured to be executed. As per our trigger configuration, it will be automatically executed on any “Pull Request” on our application repository.

8.1.4 Executing the Developer Workflow

To demonstrate the developer workflow execution, perform the following steps:

1. Switch to the application repository.
2. Make some code change and create a pull request against the “main” branch. This should automatically trigger our Code Risk Analyzer pipeline on IBM Cloud. You can view the status of our pipeline execution in our configured pipeline.

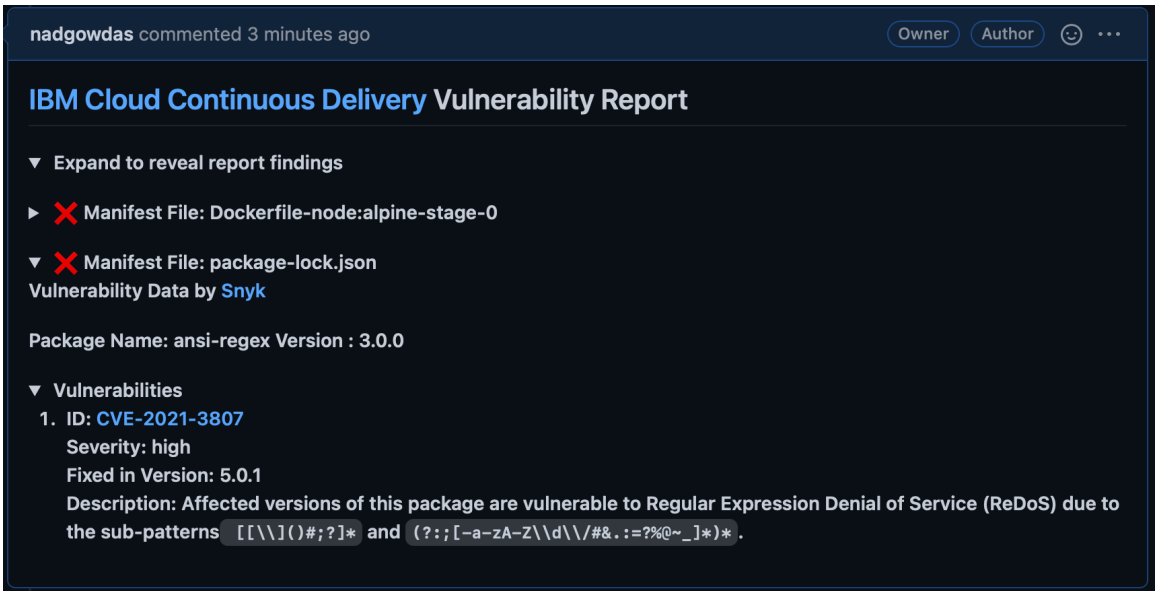


3. Once all these pipeline tasks finish, they emit their findings as git-comments to your original pull request.

8.1.5 Reviewing the Code Risk Analyzer Results

After successful execution of our pipeline, you can find the following updates in your pull requests:

- **Deployment Configuration Analysis**. If there are any Kubernetes deployment manifests (*.yaml) in the code repository, they are scanned against Docker CIS Benchmarks to identify any failures to follow best practices.
- **Vulnerability Report**. The Code Risk Analyzer allows you to discover vulnerabilities in your application (Python, Node.js, Java, golang) and OS stack (base image) based on rich threat intelligence from Snyk. It also provides fix recommendations, as the example below illustrates.



- **Bill of Materials**. The Code Risk Analyzer generates a Bill of Materials (BoM) accounting for all the dependencies and their sources for your application. The BoM is produced in JSON format. The image below shows a portion of a BoM example.

```
▼ root:
    timestamp: "2021-10-04 18:16:22"
    giturl: "https://github.com/nadgowdas/hello-containers"
    gitbranch: "nadgowdas-patch-2"
    commit_id: "74de01554d958f785ab1ee15a67850f17f19c12a"
    evidence_type: "gitsecure_bill_of_material"
    description: "A bill-of-materials report generated by IBM Code Risk Analyzer"
  ▼ build_assets: [] 1 item
    ▼ 0:
        manifest: "Dockerfile"
      ▼ stage: [] 1 item
        ▼ 0:
            stage_name: "stage-0"
            start_line: 1
            end_line: 12
          ▼ base_image:
              id: 1026331
              name: "node"
              tag: "alpine"
              os_name: "alpine"
              os_version: "3.13.6"
              sha256: "sha256:a3c0a72e086ae7e73b6742b36bc9016c27f707801744aefdd4e316ffa693bbfc"
            ▼ os-packages: [] 15 items
              ▼ 0:
                  name: "libcrypto1.1"
                  version: "1.1.1l-r0"
                  ecosystem: "os"
              ▼ 1:
                  name: "musl"
                  version: "1.2.2-r1"
                  ecosystem: "os"
                  source: "https"
              ▼ 2:
                  name: "apk-tools"
                  version: "2.12.7-r0"
                  ecosystem: "os"
              ▼ 3:
                  name: "ca-certificates-bundle"
                  version: "20191127-r5"
                  ecosystem: "os"
                  source: "https"
```


- **Git Status**. In addition to git comments describing the security findings, Code Risk Analyzer also assigns Pass/Fail status to the pull request. This allows the application owner to enforce policy-based gates to automatically block code changes with security failures.
- **Terraform Scan**. Terraform is frequently used to define and configure cloud-based infrastructure for proper application deployment. The Code Risk Analyzer also scans any terraform provider files to detect compliance issues before actual deployment. Examples of compliance checks include requirements for the minimum strength of passwords, and identity and access management (IAM) requirements for users and services. Code Risk Analyzer supports the configuration of a profile for terraform scans; this enables choosing rule parameters and which rules to run. An embedded JSON file in the git repo can contain the following properties, which are also illustrated in the screenshot below:

- "scc_goals" - SCC goals to evaluate by goal ID
- "scc_goal_parameters" - Parameter values for configurable SCC goals

```
{
  "scc_goals": [
        { "scc_goal_id": "3000010" },
        { "scc_goal_id": "3000015" }
  ],
  "scc_goal_parameters": {
    "no_of_managers_for_cloudant_db": 4
  }
}
```

8.2 IBM MaaS360 with Watson Phase 1

IBM MaaS360 with Watson is a cloud-based platform that enterprises can utilize for enterprise mobile device management (MDM) and desktop/laptop management. MaaS360 allows users to enroll organization-owned and personal devices. Administrators can send enrollment requests to users, centrally manage security policies, wipe corporate data, and push apps to devices. IBM MaaS360 is operated using an online portal. The platform system requirements for IBM MaaS360 components like client device OSs and web browsers are listed here.

Our build used this service for asset identification and assessment, routine patching and emergency patching, emergency workarounds, and isolation of assets that cannot be patched. For the first phase of our build, our managed devices were a MacBook Pro and a Windows 10 virtual desktop. For the second phase of this project, Android and iOS mobile devices were managed.

MaaS360 provides a Quick Start guide for customers when logging in for the first time. The guide helps with setting up device enrollment, establishing security policies, configuring corporate email, and enrolling devices. For this lab, a corporate identifier was set up, as well as an internal AD (lab.nccoe.org), as the default identification mode. The corporate identifier allows users to enroll their devices in MaaS360. More information on configuration can be found here.

8.2.1 Enrolling Devices

IBM MaaS360 supports traditional endpoints running Windows 10 and up, as well as macOS endpoints. Device enrollment is critical in helping enterprises register devices and apply device management policies that are specific to their organization. The IBM MaaS360 Portal handles all enrollment settings for devices, apps, and users.

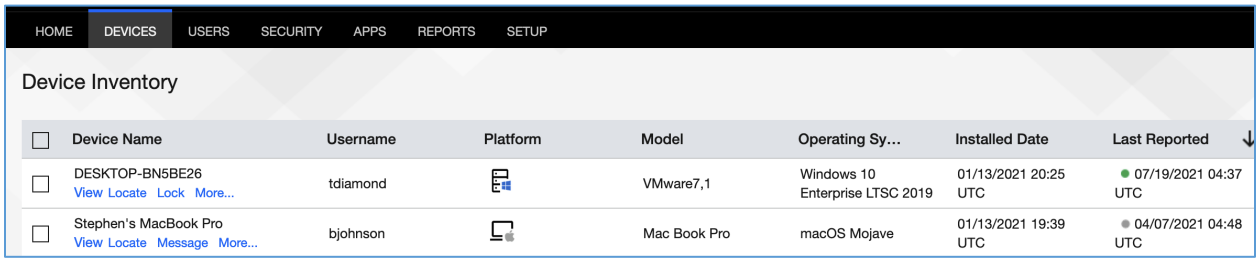
The device enrollment process is as follows:

1. Select **SETUP**, then choose **Settings**.

2. Click **Device Enrollment Settings** and set the corporate identifier that users will utilize to enroll devices. It will also be shown in the enrollment URLs. In our lab, we set the identifier to "nccoelab". Additional device enrollment settings can be found here.
3. Open a web browser and proceed to the MaaS360 enrollment URL.
4. Enter your credentials.
5. Review and accept the terms and conditions.
6. Install the MDM profile.

Figure 8-1 provides a sample of enrolled devices utilized in this lab.

Figure 8-1 Sample of Enrolled Devices



See the linked pages for detailed enrollment instructions, including bulk enrollment, for macOS devices and Windows devices.

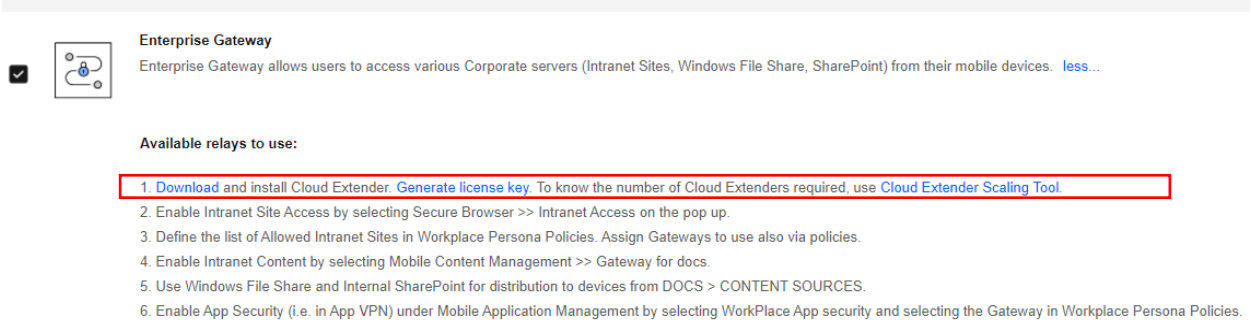
8.2.2 Cloud Extender Installation

The IBM MaaS360 Cloud Extender allows enterprises to integrate mobile devices with corporate on-premises and cloud-based resources. The Cloud Extender is installed on a Microsoft Windows server behind the firewall to allow users and devices to use internal resources like directory services, file shares, email, and applications.

In this lab, the Cloud Extender was installed on the AD server to allow users to log in with AD accounts. The Maas360 portal provided links to the Cloud Extender software download, installation, and license key generation; they were available on the **SETUP** menu under **Enterprise Gateway**, as Figure 8-2 shows. The same line also pointed to a scaling tool that can aid administrators in calculating the number of Cloud Extenders needed.

More information about the requirements and instructions on installing the Cloud Extender can be found here.

2236 **Figure 8-2 IBM Maas360 Cloud Extender Download**



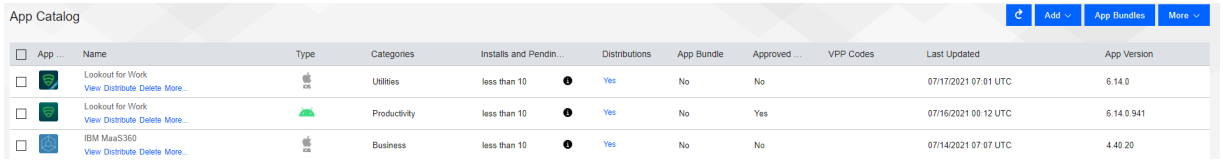
2237 8.2.3 App Catalog and Distribution

2238 This lab build sought to demonstrate how a tool like MaaS360 could be helpful in allowing
2239 administrators to more easily distribute applications required for business operations to users. MaaS360
2240 provides the ability to push applications to users by allowing administrators to build a customizable app
2241 catalog. An app catalog makes it easier to distribute custom apps created by the organization. Also,
2242 multiple versions of the same app can be pre-released to specific groups as a test before a full
2243 deployment.

2244 Lastly, the app catalog allows for the remote distribution, installation, uninstallation, updating, and
2245 configuring of applications. The ability to remotely control apps is an important step in managing
2246 updates and security for an enterprise's patch management process. In MaaS360, applications must be
2247 added to the app catalog before they can be deployed to devices. The following outlines the steps:

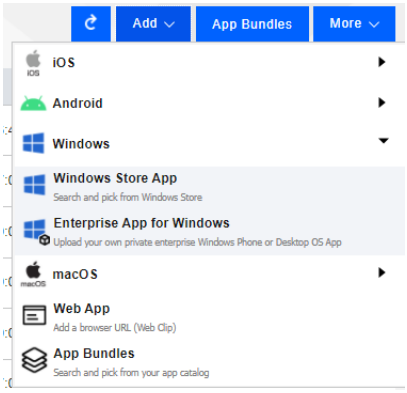
2248 1. From the Maas360 Portal's **APPS** menu, select **Catalog**.

2249 2. The image below is a sample App Catalog page from this project's build. To add an application,
2250 click **Add**.



2251

2252 3. Next, select the kind of app that will be added. For this example, **Windows Store App** is
2253 selected.



4. Enter the desired app in the **App** field (for example, Slack) and select **Add** to complete the process.

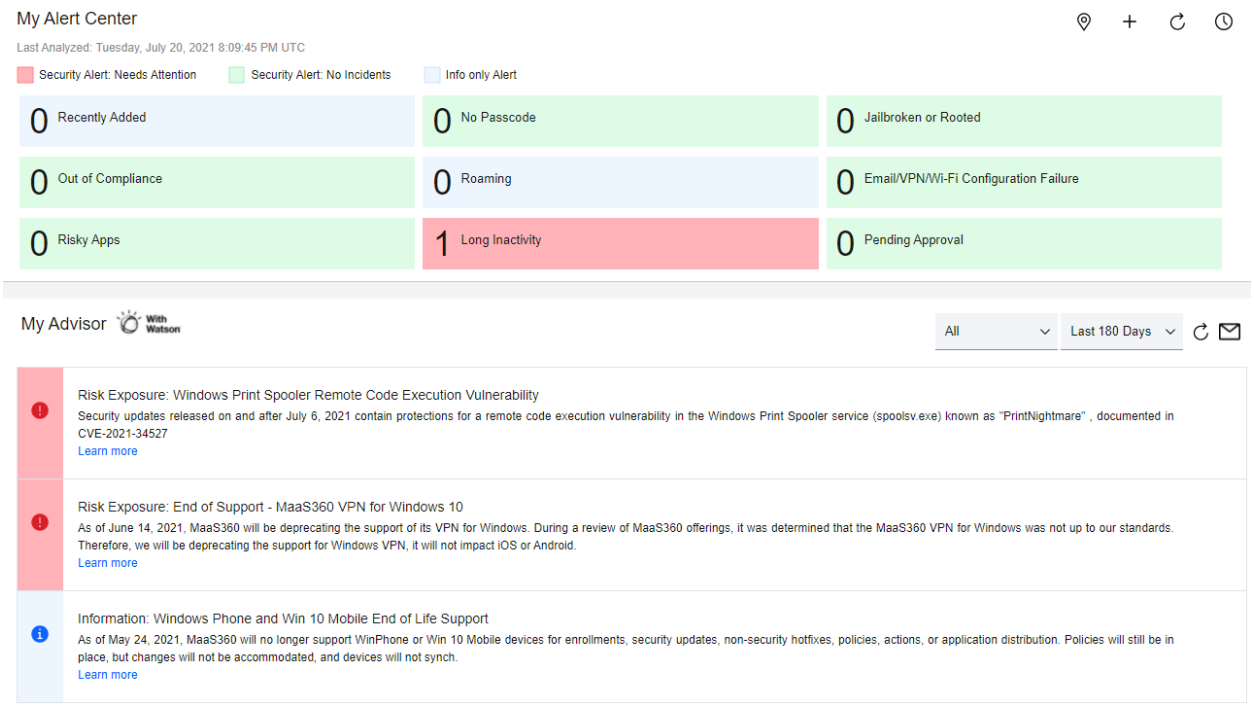
To distribute an app after it is added to the app catalog, select it from the App Catalog list, then click **Distribute**.

Additional information on the app catalog and app installation can be found here.

8.2.4 Deploying Patches

This build utilized the capability of MaaS360 to provide alerts about required patches and take action to remedy the issues. MaaS360 listed alerts on a dashboard on the Home Page, as illustrated in Figure 8-3. The first half of the page utilizes colored tiles to demonstrate items in compliance (green) and those that need attention (red). The information listed on these tiles can be customized. Below the security alert tiles, there is a My Advisor with Watson section. Watson is an artificial intelligence tool developed by IBM that scans the internet and other resources for the most recent trends in malware. It then lists any threats found that are linked to devices that are enrolled in MaaS360. Additional information about the MaaS360 Portal Home Page is listed here.

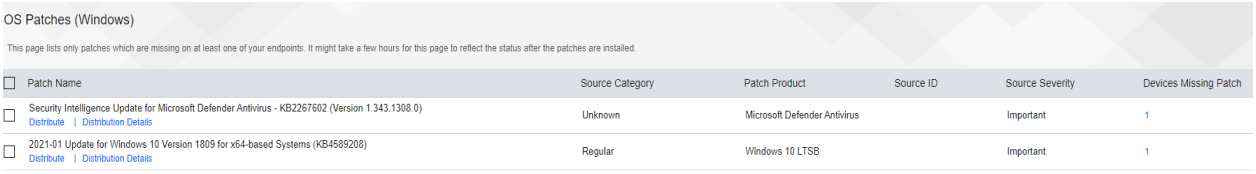
2269 **Figure 8-3 MaaS360 Portal Home Page**



2270 MaaS360 allows administrators to apply and distribute patches to a single device or multiple devices.
2271 The patches page for Windows and macOS devices lists the patches that are missing from devices.
2272 Administrators are also able to see current patching schedules. For example, these steps were followed
2273 in the lab to use MaaS360 to schedule deployment of patches:

2274 1. Select **SECURITY**, then click **OS Patches (Windows)**.

2275 2. Select the Windows machines to be patched, then click **Distribute** to apply the patches.



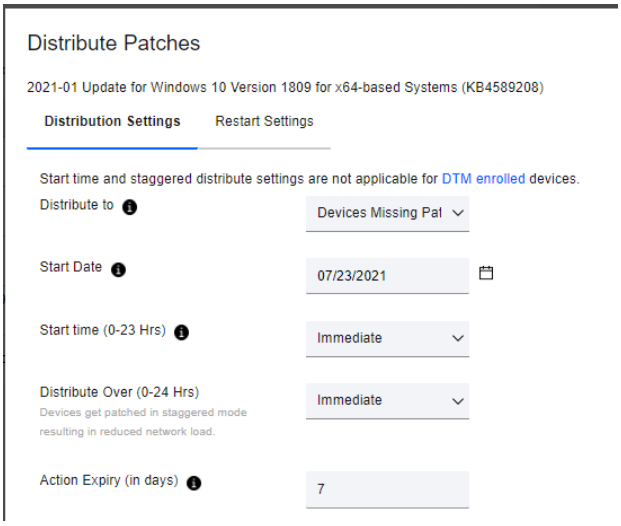
2276 The options shown in Table 8-1 were utilized to schedule automated patching.

2277 **Table 8-1 Values Specified for Scheduling Automated Patching**

Distribution Setting	Value	Explanation
Distribute to	Devices Missing Patches	Choose which device(s) to apply patching to. Select from **Single Device**, **Device Groups**, or **All Devices**.
Start Date	12/04/2021	This field sets the date that the remediation step will happen. This field was set to a future date so that patches would be scheduled for deployment.
Start Time (0-23 Hrs)	01	This establishes the time of day that distribution will start for selected devices.
Distribute Over (0-24 Hrs)	15	This causes patching to be staggered to reduce network load by making updates available over a set amount of time, in this case 15 hours, instead of instantaneous availability for all users.
Action Expiry (in days)	7	The action will automatically expire after seven days.

2278 To distribute patches out of schedule for emergency remediation needs, the following options were
2279 utilized:

2280 - Start Date: The current date was chosen
2281 - Start Time (0-23 Hrs): Immediate (causes the patch to be deployed immediately)
2282 - Distribute Over (0-24 Hrs): Immediate
2283 - Action Expiry (in days): 7



2284

More information about patch management with IBM MaaS360 is available here.

8.2.5 MaaS360 Maintenance

MaaS360 is a SaaS offering, so updates are continuously pushed out by IBM, who maintains the platform.

8.3 IBM MaaS360 with Watson Phase 2

This section goes over phase 2 deployment of the lab instance utilizing MaaS360. The phase 2 build utilized MaaS360 to administer Google Android and Apple iOS devices.

8.3.1 Enrolling Mobile Devices

In our build we enrolled mobile devices in both a supervised state (or corporate owned) and a BYOD method. Corporate owned or supervised status means that organizations have full control over the device, as opposed to BYOD where organizations only have control over the work applications on the device. The following is an overview of how to enroll an iOS device in a supervised state using the Apple configurator:

1. Select **Devices** > **Enrollments** > **Other Enrollment Options** and select **Apple Configurator** from the dropdown menu.
2. Select the **Non-DEP only Enrollment** URL (Note: DEP stands for Device Enrollment Program), and copy the URL from the **With Authentication** tab.
3. Connect the Apple iPhone or iPad device to a MacBook and start the Apple Configurator.
4. Follow the wizard through specifying the MDM Server URL and certificates, and assigning the device to an organization.

For more information on enrolling iOS devices in MaaS360 using Apple Configurator, review the following page.

The lab instance enrolled Android devices manually using a QR code during device setup. The following steps provide an overview for how to do this:

1. Click on **Devices** > **Enrollments** > **Other Enrollment Options** > **Android Enterprise QR Code Provisioning**.
2. Enter the requested information into the form that appears. Of note is the Android Enterprise Mode options for Android Enterprise mode. The **Device Owner** option allows an organization to have complete control over the device, while **Work Profile on Corporate Owned** allows organizations to only manage apps under the work profile of the device.

3. During initial setup of the mobile device, tap the screen six times and then scan the QR code displayed in the MaaS360 portal.

For more information on enrolling devices using a QR code, follow the information on this page.

While the lab instance utilized manual processes to enroll devices, MaaS360 also supports bulk enrollment of Apple and Google devices. For information, consult the following links:

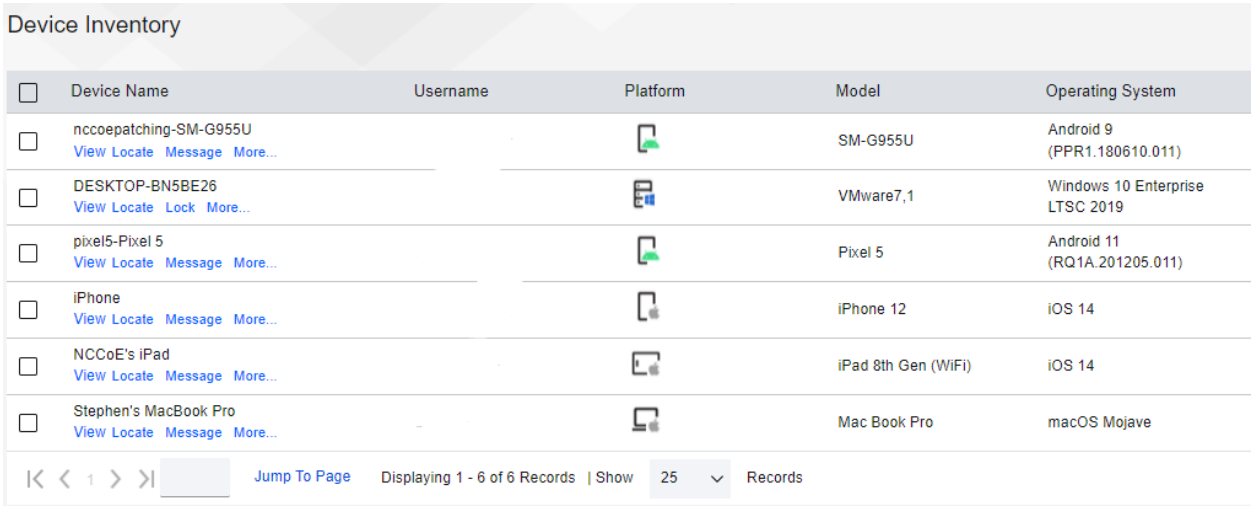
- Bulk Enrolling Android devices using Android Zero Touch enrollment
- Bulk Enrolling Apple devices using Apple Device Enrollment Program

8.3.2 Device Inventory

The **Devices** > **Inventory** tab lists all the devices that have been enrolled into MaaS360. The patching instance utilized this tab to perform firmware and software discovery capabilities.

The firmware of enrolled devices is displayed directly on the device inventory list. The **Operating System** field shows the detected OS on the device. Figure 8-4 shows the connected devices in the patching instance with the OSes that were detected.

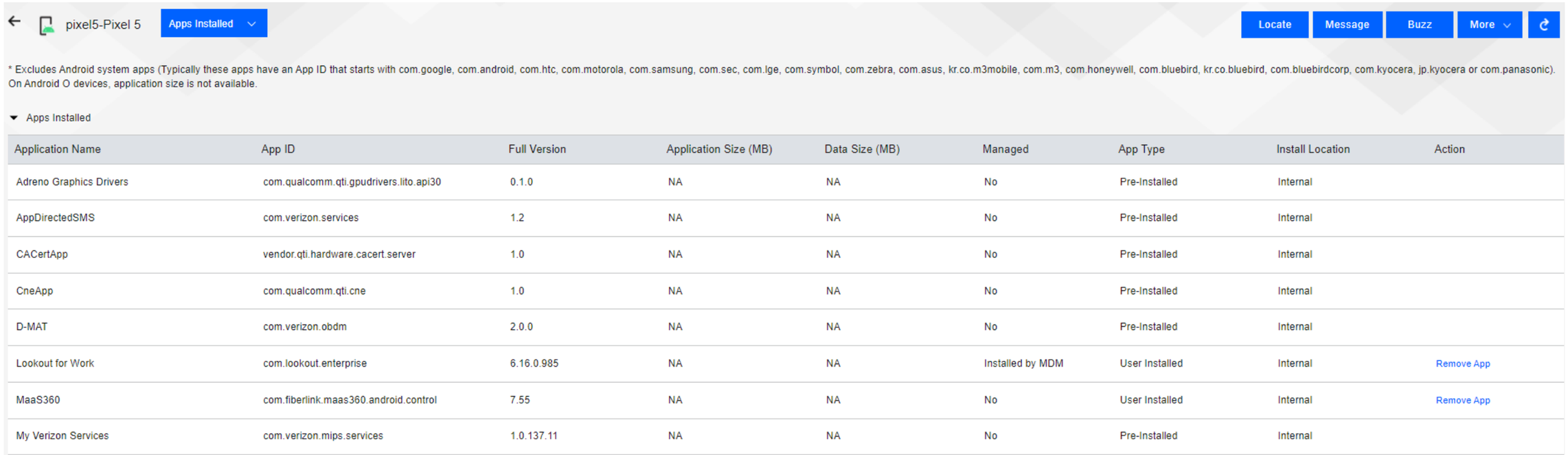
Figure 8-4 Example of Enrolled Device Inventory



More detailed information regarding the installed OS and hardware information can be found under **View** > **Device Summary** > **Hardware & OS**.

The installed applications on enrolled devices can be found by going to **Device Inventory** > **View** > **Apps Installed**. The Apps Installed list shows all installed applications on a device. Figure 8-5 gives an example of installed applications on a device. The list allows user-installed applications to be uninstalled by administrators.

2335 **Figure 8-5 Example of Installed Apps on a Mobile Device**



Application Name	App ID	Full Version	Application Size (MB)	Data Size (MB)	Managed	App Type	Install Location	Action
Adreno Graphics Drivers	com.qualcomm.qti.gpudrivers.lito.api30	0.1.0	NA	NA	No	Pre-Installed	Internal	
AppDirectedSMS	com.verizon.services	1.2	NA	NA	No	Pre-Installed	Internal	
CACertApp	vendor.qti.hardware.cacert.server	1.0	NA	NA	No	Pre-Installed	Internal	
CneApp	com.qualcomm.qti.cne	1.0	NA	NA	No	Pre-Installed	Internal	
D-MAT	com.verizon.obdm	2.0.0	NA	NA	No	Pre-Installed	Internal	
Lookout for Work	com.lookout.enterprise	6.16.0.985	NA	NA	Installed by MDM	User Installed	Internal	Remove App
MaaS360	com.fiberlink.maas360.android.control	7.55	NA	NA	No	User Installed	Internal	Remove App
My Verizon Services	com.verizon.mips.services	1.0.137.11	NA	NA	No	Pre-Installed	Internal	

2336 For more information regarding the device inventory page, please consult the following page.

2337

8.3.3 Device Policies

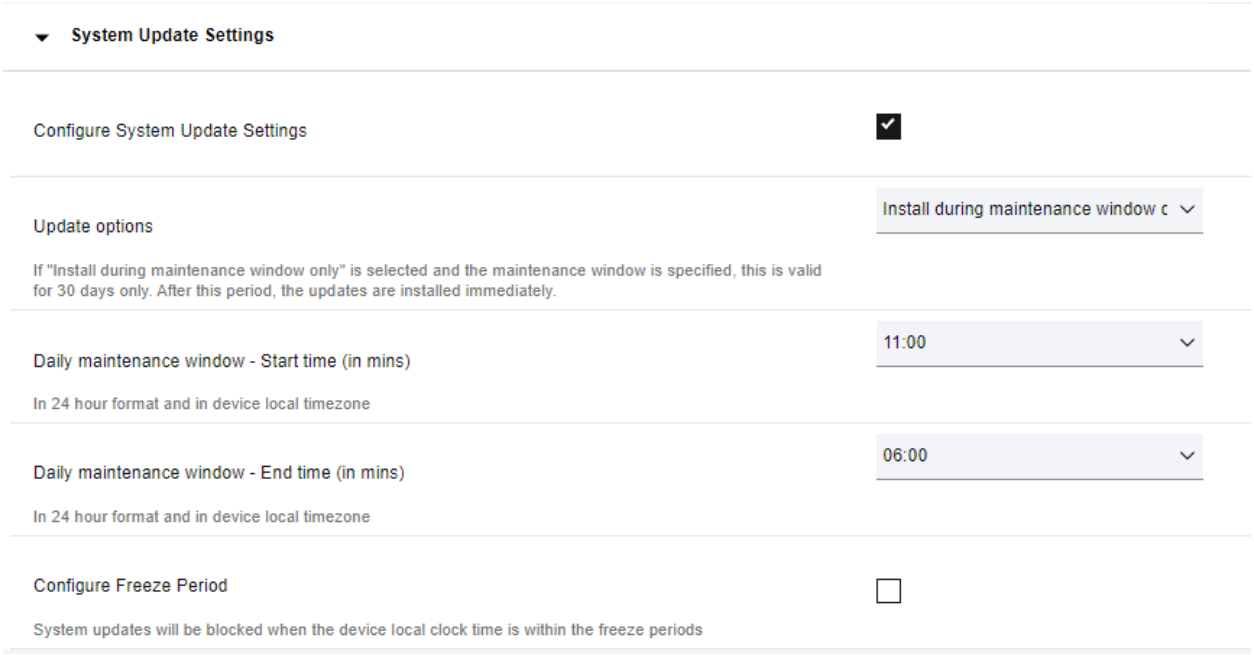
The **Security** > **Policies** section of MaaS360 allows security policies and settings to be applied to devices to make sure they comply with organizational policies. The lab instance utilized policies to ensure that the Android devices had the MaaS360 and Lookout for Work applications, as well as to perform automatic OS updates. Policies for Apple were configured to require the MaaS360 and Lookout for Work applications.

To configure the Android Default Policy to support required apps and automatic updates, perform the following steps:

1. Click **Security** > **Policies**.
2. Under **Default Android MDM Policy**, click **View**.
3. Under the **Android Enterprise Settings** field, click **App Compliance**.
4. Click the **Edit** button.
5. Select **Configure Required Apps**.
6. Under **Application Name**, type the following:

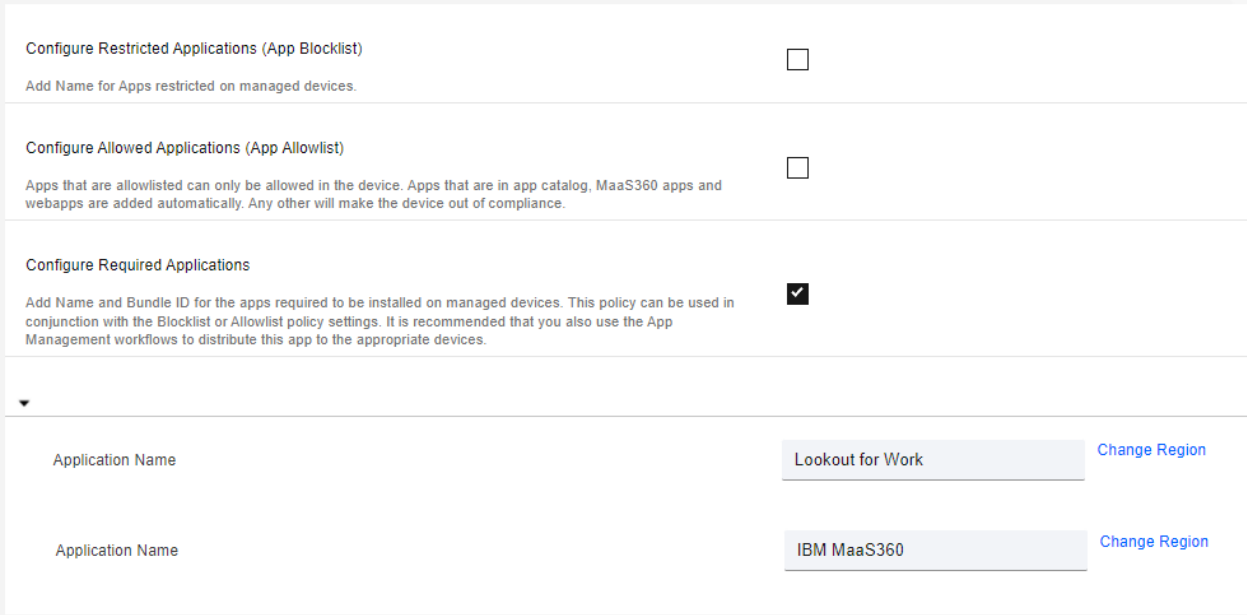
```
app:com.fiberlink.maas360.android.control
app:com.lookout.enterprise
```

7. Click on **Android Enterprise Settings** > **System Update Settings**.
8. Click the check box for **Configure System Update Settings** and fill out the information as shown in the screenshot below.



To configure a default Apple policy to require the installation of MaaS360 and Lookout for Work, perform the following steps:

1. Click **Security** > **Policies**.
2. Click on the **Default iOS MDM Policy**.
3. Under **Device Settings**, click on **Application Compliance**.
4. Click **Edit** and check the **Configured Required Applications** check box.
5. Under the application names, add **Lookout for Work** and **IBM MaaS360**. Note that typing in the Application Name field will cause MaaS360 to search for the application. Click the application when it appears in the search field.



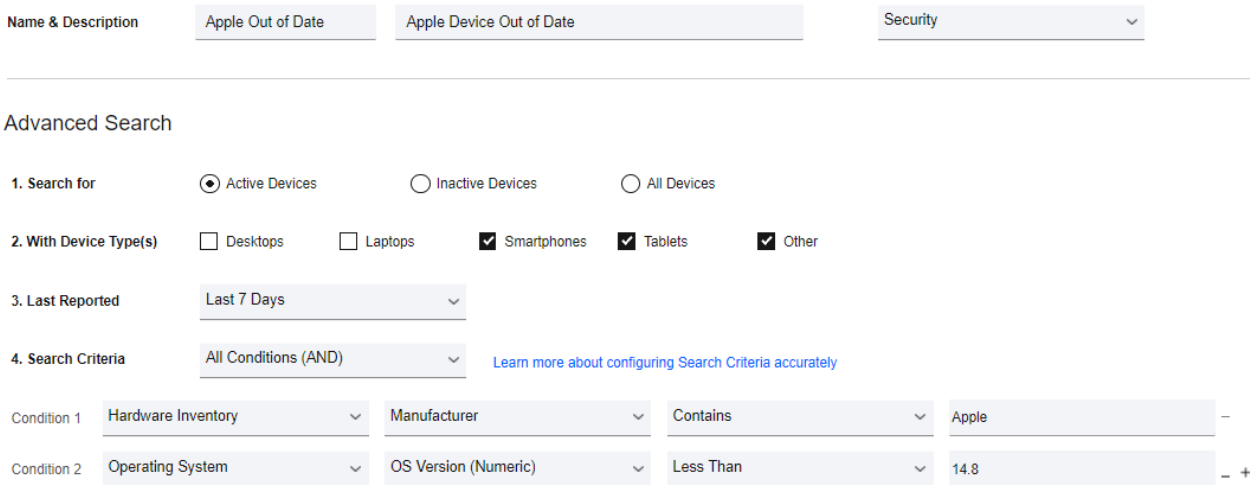
6. Click the **Save And Publish** button.

8.3.4 Alerts

MaaS360 can alert via the My Alert Center dashboard on the home page. The My Alert Center dashboard can have custom alerts added. The lab instance used this capability to alert administrators when mobile devices might be running older firmware versions. For more information on building the alert center, see the following page.

The following steps walk through creating an alert to show out-of-date Apple devices:

1. Click the **Home** tab of MaaS360.
2. Under the My Alert Center, click the plus icon.
3. Fill out the **Search Criteria** as shown below. This rule creates a search that looks for devices that are currently active and fall under the category of smartphones and tablets. To make sure that we only look at Apple devices, the first condition has been set to match devices that have been manufactured by Apple. The second condition sets the OS version to be no less than 14.8.



4. Click **Save**.

5. The My Alert Center will update with the results of the search, as shown below.

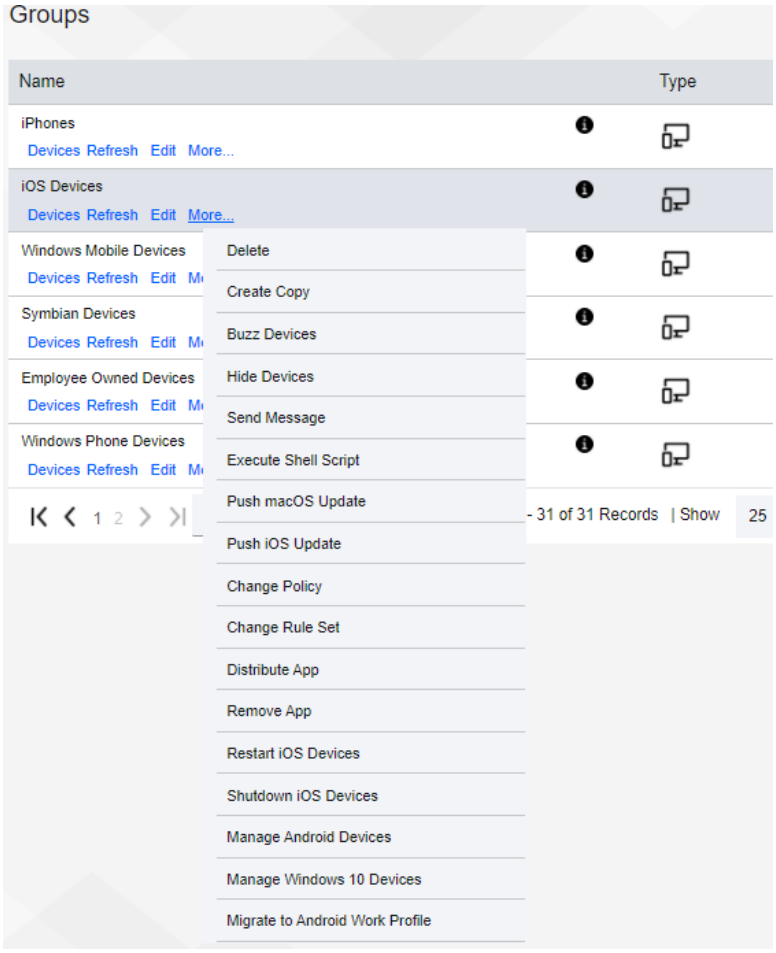
8.3.5 Firmware Updates

The MaaS360 tool can push out firmware updates to devices that are corporate owned or enrolled as supervised devices. The lab instance utilized this feature to push firmware updates to devices to meet the firmware patching scenario.

Android device patching was covered in Section 8.3.3. The policy that was previously configured will have Android devices automatically install software updates during a defined maintenance window. Administrators can set the policy to automatically install updates as soon as they are available instead of waiting for a maintenance window. This can be used to provide immediate patching in the case of emergencies. Please consult the following page for more information on configuring policy for Android system updates.

Apple iOS devices do not have a way for policy to be configured to automatically push out system updates. However, administrators can still push out iOS updates to supervised devices through a manual process. To push out an Apple iOS update to a group of devices, perform the following steps:

1. From the MaaS360 Portal click on **Devices** > **Groups**.
2. Under the **Groups** list, find iOS devices. Note that other device groups are automatically available, such as iPhones or iPads, if an administrator does not want to push out the update to all iOS devices.
3. Under the **More** button, select **Push iOS Update**.



4. The Push iOS Update window will appear. Select **Download and Install**, then click the **Continue** button.

8.4 IBM MaaS360 with Watson Reporting

IBM MaaS360 has the capability to create a variety of reports that may help administrators gain better insight of the enterprise's mobile environment. Reports are available for hardware inventory, network, app inventory, mobile data usage, user endpoint management overview, and app security settings. Administrators can also customize reports and opt to have reports delivered on a daily, weekly, or monthly basis. Reports are refreshed every 24 hours, and they are available for data that is up to 180 days old. There are also filters available that may be helpful with managing the report data.

Reports can be accessed by selecting **REPORTS** from the MaaS360 Portal, then choosing the type of report that is needed. For example, the sample report from the lab shown in Figure 8-6 broke down devices by platform to provide an asset inventory.

Figure 8-6 Sample Report from MaaS360

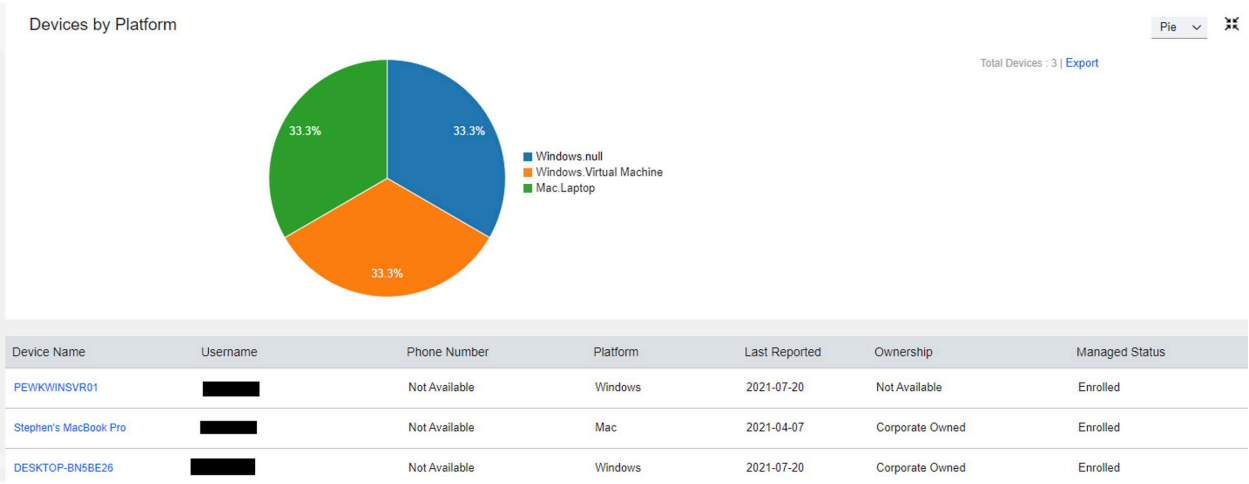
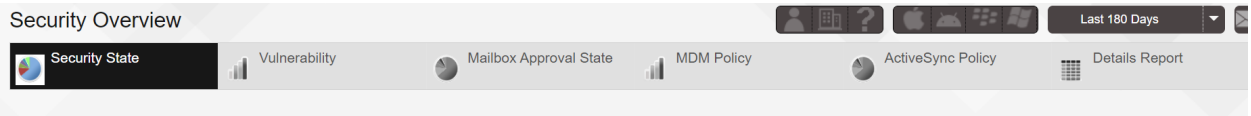


Figure 8-7 demonstrates the administrator's ability to create reports based on the Security State, Vulnerability, Mailbox Approval State, MDM Policy, ActiveSync Policy, and Details Report.

Figure 8-7 IBM Maas360 Report Options



Additional instructions concerning managing reports in MaaS360 are available here.

9 Lookout

Lookout Mobile Endpoint Security (MES) is a SaaS-based mobile threat defense solution that collects information from devices via the Lookout for Work mobile application. In our build, it provided vulnerability scanning, assessment, and reporting for Android and Apple mobile devices. Also, we integrated Lookout MES with IBM MaaS360 to provide security policy enforcement actions.

9.1 Integrating Lookout with IBM MaaS360

Lookout MES device enrollment can be accomplished without third-party integration. However, to enforce installation, the Lookout for Work client must be managed and pushed via an MDM to mobile devices. In our build, the MDM was IBM MaaS360 with Watson. Detailed information regarding integrating Lookout and MaaS360 can be found here. Please note that you will need an account to view the documentation. The following steps provide a high-level overview of integrating Lookout MES with MaaS360:

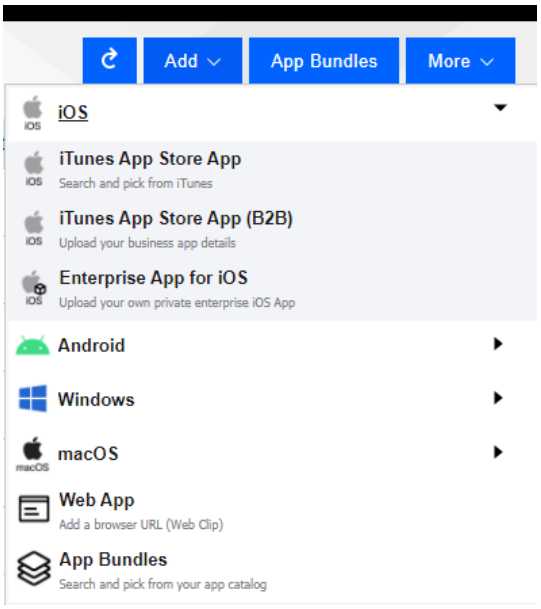
1. Create an API user in MaaS360: This step creates a user in MaaS360 with the correct permissions that can then be used for Lookout MES to access the MaaS360 API.
2. Create custom attributes in MaaS360: Lookout MES passes device state information back to MaaS360. Custom attributes will need to be set up in MaaS360 so that the information passed by Lookout can be stored by MaaS360 and used in policy enforcement. The following attributes are created:
   - lookout_activation_state: This specifies whether the Lookout for Work app is installed and activated on the device.
   - lookout_device_state: This indicates the overall state of the device, such as secured, threats detected, deactivated, or pending activation.
   - lookout_disconnected: This indicates if there is a connection from the mobile device to Lookout.
   - lookout_threat_level: This categorizes the threat level of the device by none, low, medium, or high.
   - lookout_unreachable: This indicates if the Lookout MES server is reachable by the mobile device.
3. Add the Lookout for Work app to the MaaS360 App Catalog.
4. Configure the MaaS360 connector from the Lookout Console.

9.2 Adding Lookout for Work to the MaaS360 App Catalog

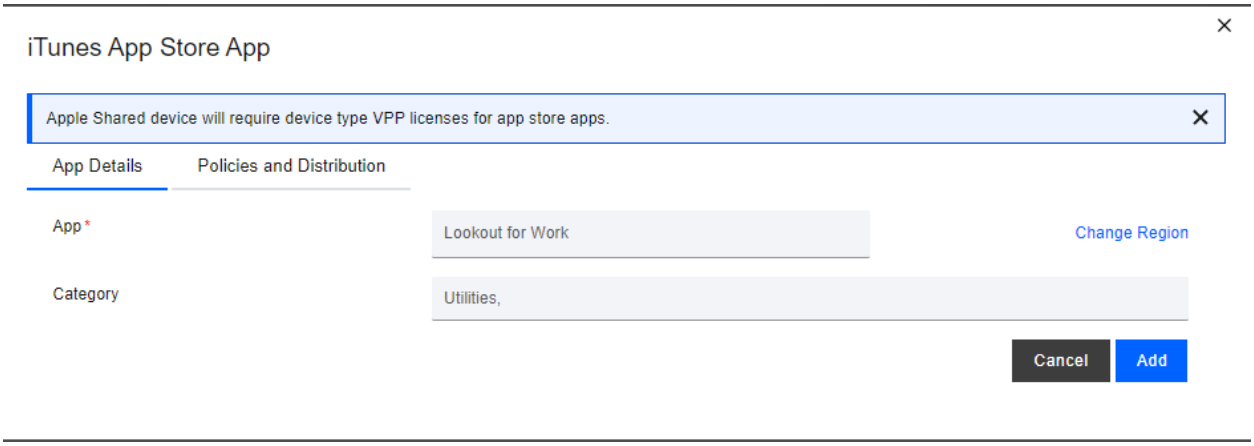
Adding the Lookout for Work iOS and Android applications to the MaaS360 App makes the application available in the IBM MaaS360 app store. For supervised or corporate-owned devices, the application will install automatically without further user interaction. More information for adding the Lookout for Work App to MaaS360 can be found here.

The following steps provide an overview of the process of adding Lookout to MaaS360:

1. From the MaaS360 Portal, select **APPS** and then click **Catalog**.
2. Click **Add** and choose the OS required (**iOS** is chosen for this example).



3. Next, select iTunes App Store App. Then enter *Lookout Mobile Security* in the search bar and click **Add**. An example is found below.

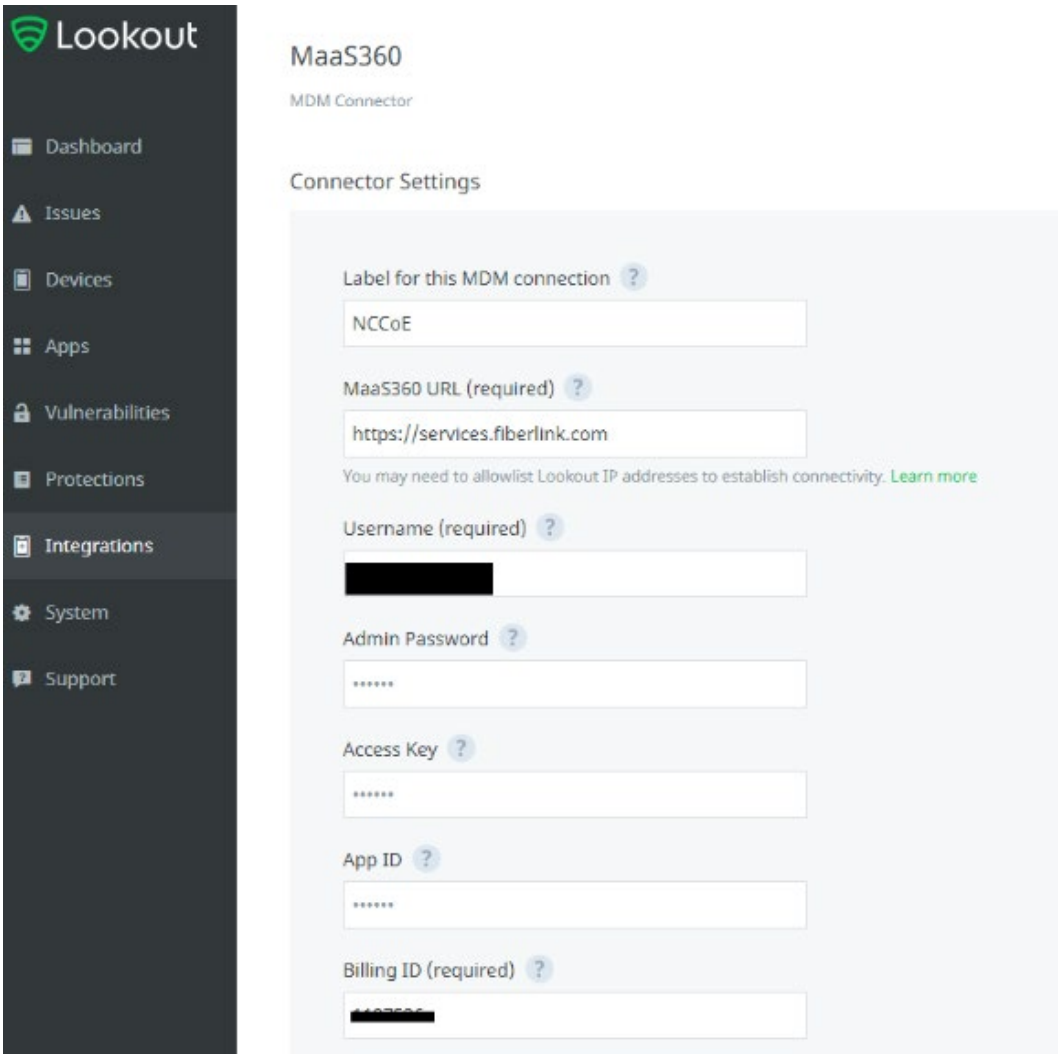


4. Add the Lookout for Work configuration details so that when users open the application, it will be automatically configured and connect to Lookout without further interaction.

9.3 Configuring MaaS360 Connector in the Lookout MES Console

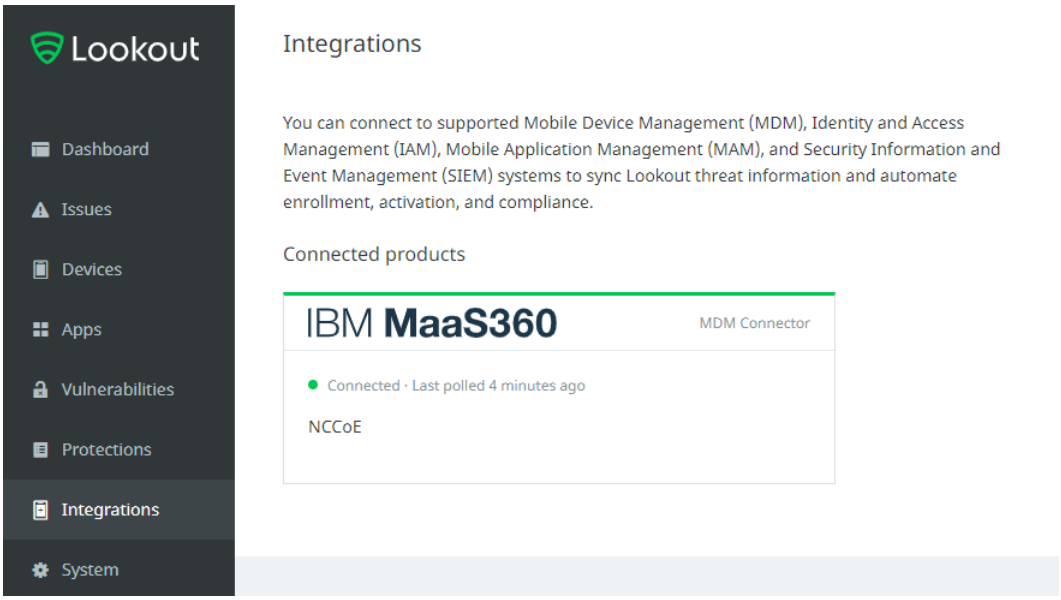
To integrate Lookout MES with IBM MaaS360, perform the following steps:

1. Select **Integrations** in the Lookout MES console.
2. Enter the Label for the connection, MaaS360 URL, the API username and password, Access Key, Apple ID, and Billing ID. An example is shown below.



2471

2472 After a successful integration, the Integration page should display the following:

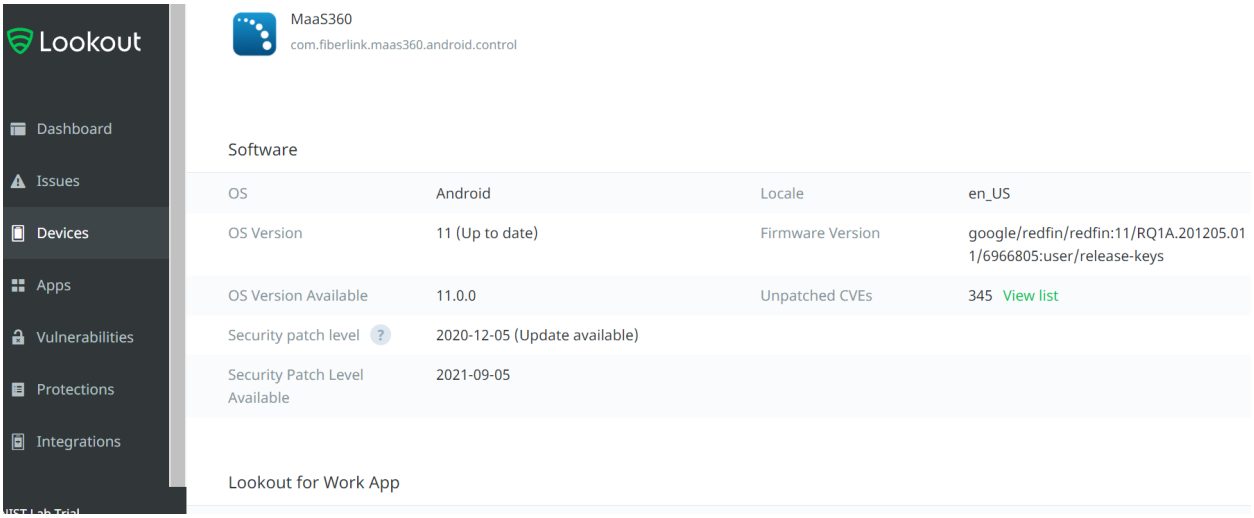


9.4 Firmware Discovery and Assessment

Once Lookout for Work is activated, it collects details about devices that include the device's OS version and the patch level for Android devices, and then lists all CVEs associated with the device based on the OS version and Android Security Patch Level (ASPL). The Lookout MES platform can discover firmware (the OS running), and it displays this information under the **Devices** tab.

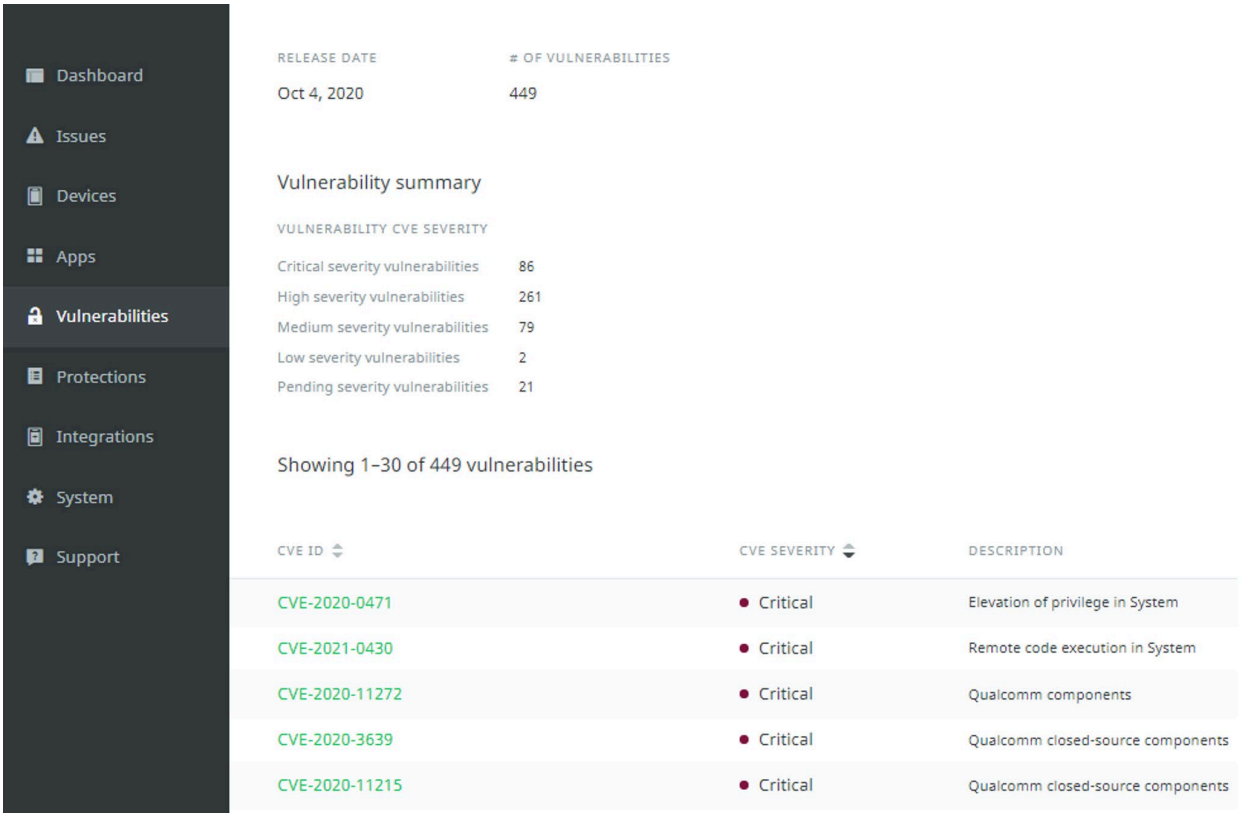
Once the **Devices** tab is chosen, a list of all connected devices are displayed in the window. Select a device from the list to discover its firmware. Then information about the device's firmware, including OS and Security Patch level, can be found by scrolling down to the software section. An example of this information is displayed in Figure 9-1.

2483 **Figure 9-1 Example of Device Firmware Information**



2484 By click the **View List** button from the Unpatched CVEs section, administrators can see all CVEs that are
2485 associated with the current OS and ASPL on the device. The **Vulnerability Summary** tab breaks down the
2486 vulnerabilities associated with a device by severity. An example of this information is displayed in Figure
2487 9-2.

2488 **Figure 9-2 Example of Vulnerability Severity Information**



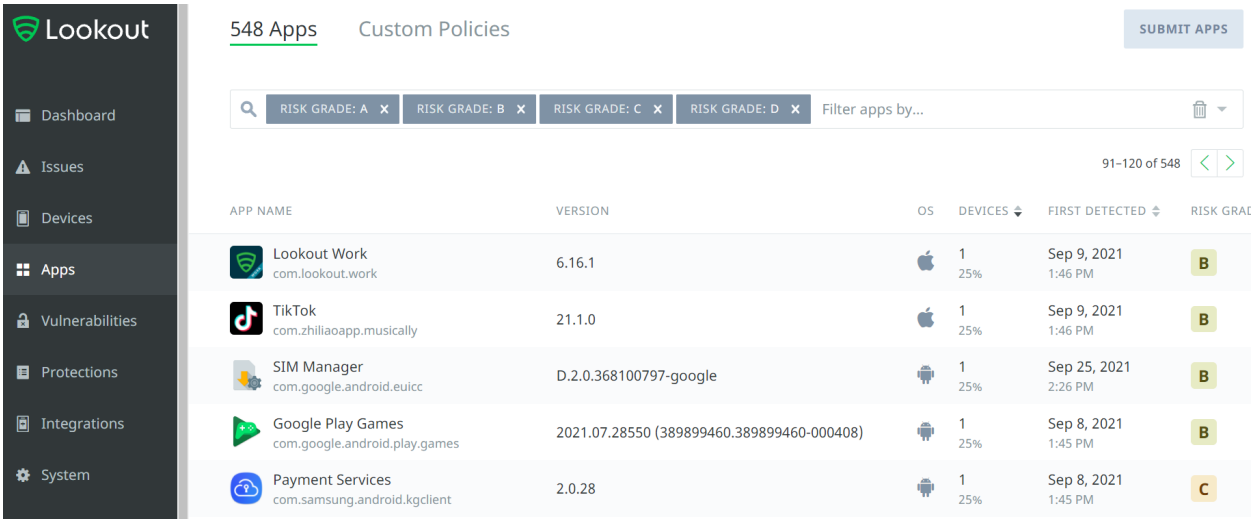
2489 9.5 Software Discovery and Assessment

2490 The activation of the Lookout for Work client allows for the collection of running applications on the
2491 device. For Android devices, Lookout collects an app inventory for the device which includes details
2492 about app versions plus libraries and software development toolkits (SDKs) used by the apps. For iOS
2493 devices, this information is obtained using the MaaS360 API. Lookout MES can also indicate if there are
2494 vulnerabilities in the applications themselves.

2495 The Lookout MES platform displays a risk grade which shows the risk that the app presents if it was
2496 compromised. Lookout calculates this grade based on the application's permission (what information it
2497 can access). Each risk grade is on an A to F scale (A, B, C, D, or F). Lookout MES does not link applications
2498 to specific devices unless a device fails a compliance check because of an installed application. For
2499 example, if there is a rule that prohibits the installation of TikTok, only devices with TikTok installed will
2500 be highlighted.

2501 To view the applications that are installed on devices, select **Apps** from the Lookout MES dashboard.
2502 Figure 9-3 shows a sample of the **Apps** page from our build.

2503 **Figure 9-3 Lookout Apps Page Sample**



2504 9.6 Lookout MES Security Protections

2505 Lookout MES allows organizations to set protection parameters for enrolled devices. Lookout comes
2506 preconfigured with multiple templated rules that can be configured to meet organizational risk
2507 tolerance. Policy enforcement can be accomplished through MES directly or via integration with an
2508 MDM.

2509 Our build utilized this feature to implement a rule to restrict network access to devices that had an out-
2510 of-date firmware level. The lab configured this rule by defining a minimum OS version and Android
2511 security patch level and by choosing to alert the device's user and block access to certain domains if the
2512 minimum is not met.

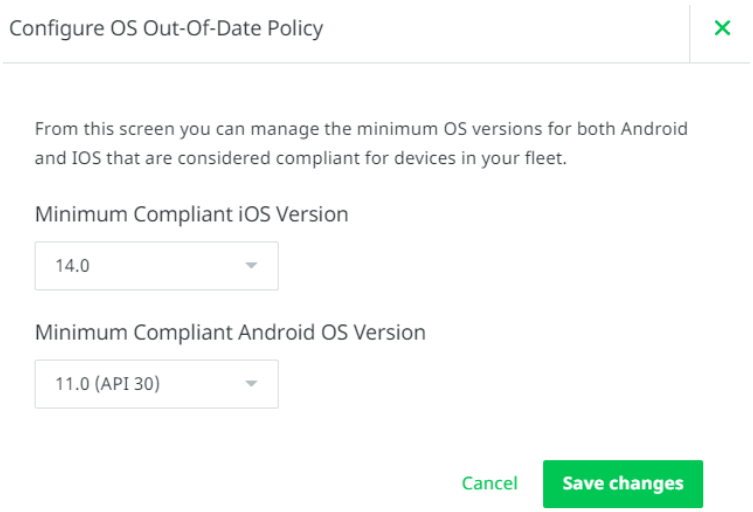
2513 To configure such a rule, perform the following steps:

2514 1. Click on the **Protection** tab.

2515 2. Scroll down to **OS Out-of-date** and select a risk level of **High** under the **Risk Level** dropdown.

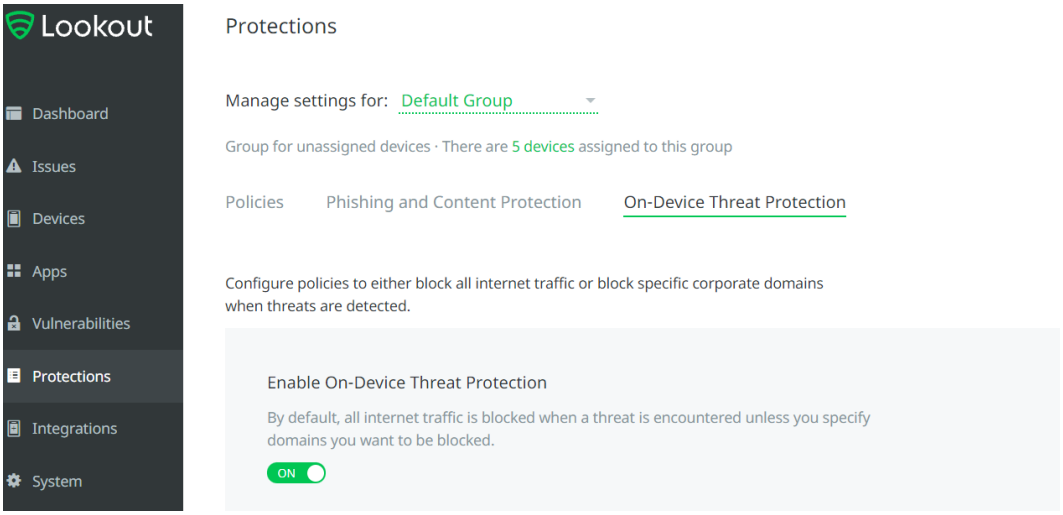
2516 3. Click the gear icon by the **Risk Level** dropdown menu.

2517 4. Select the minimum compliant iOS and Android OS versions from the dropdown, as shown
2518 below, then click **Save changes.**



2519

2520 5. From the **Protections** tab, click **On-Device Threat Protection**, and set **Enable On-Device Threat**
2521 **Protection** to **ON**.



2522

2523 6. Under the **Response** dropdown, choose **Block domains and alert devices**.

2524

2525 7. Scroll down to **Block specific domains** and click a domain to add.

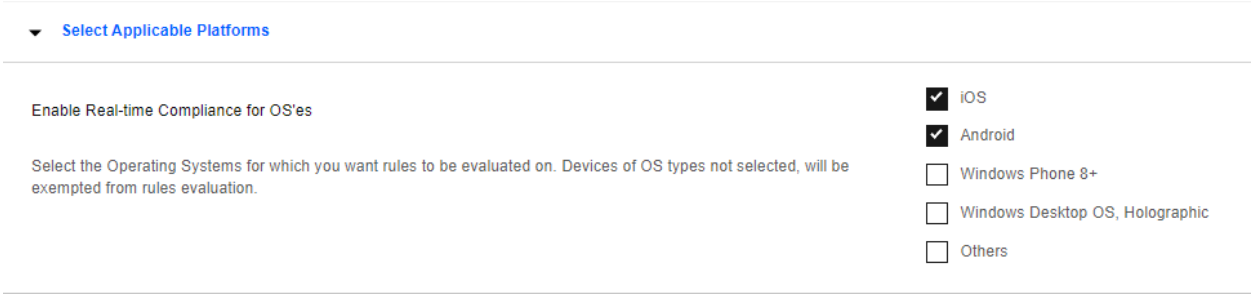
8. Specify a domain that non-compliant devices should not access. Note that domains can be added by CSV files.
9. Click **Save changes**.

9.7 Security Compliance Enforcement with IBM MaaS360

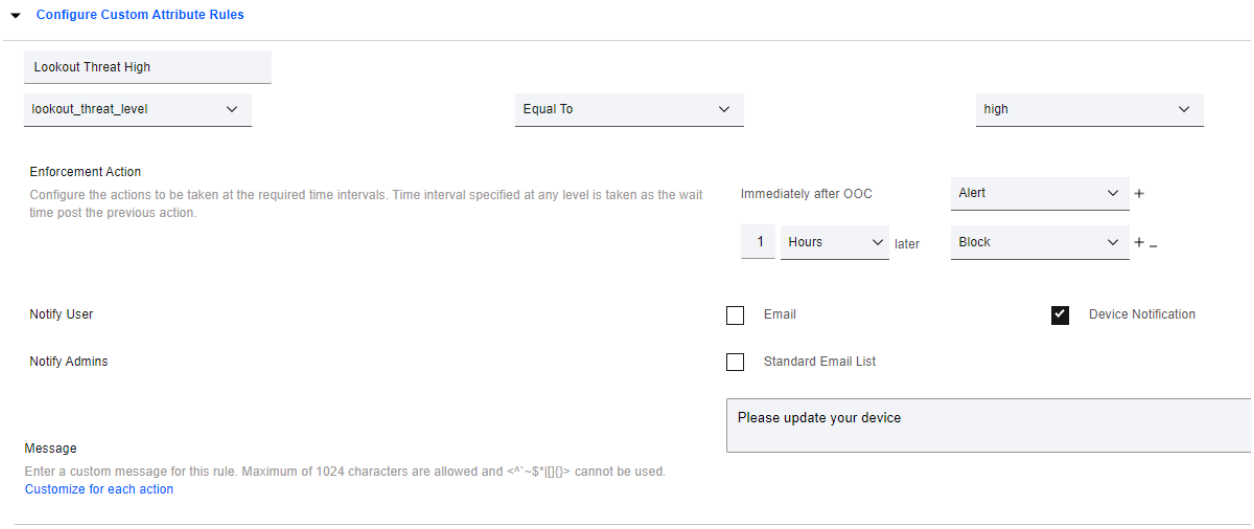
Lookout MES can pass custom attributes to MaaS360 for use in custom security compliance rules. This integration was set up in Section 9.1. Our build utilized this capability to block access to corporate resources for any device with a threat level of high by Lookout MES. Information on applying security compliance rules for devices can be found here.

The following steps show how to create a security compliance rule using Lookout custom attributes:

1. Under the MaaS360 console click **Security** > **Compliance Rules**.
2. Click **Add Rule Set**.
3. Under the **Rule Set Name Field**, type in **Lookout Custom Attributes** and then click **Continue**.
4. Under the **Basic Settings**, ensure that the **iOS** and **Android** fields are checked, as depicted below.



5. Click on **Custom Attribute Rules** and fill out the following fields:
   - Rule Name: Lookout Threat High
   - Select Attribute: lookout_threat_level (this corresponds to the threat level that Lookout assigns to a device, which was configured in section 9.6)
   - Select Criteria: Equal To
   - Choose Value: High
6. Under **Enforcement Action**, click to **Alert** and then **Block** as shown below.


Configure Custom Attribute Rules
Lookout Threat High
lookout_threat_level
Equal To
high
Enforcement Action
Configure the actions to be taken at the required time intervals. Time interval specified at any level is taken as the wait time post the previous action.
Immediately after OOC
Alert
1
Hours
later
Block
Notify User
Email
Device Notification
Notify Admins
Standard Email List
Please update your device
Message
Enter a custom message for this rule. Maximum of 1024 characters are allowed and <^`~$*|[]{}> cannot be used.
Customize for each action

7. Click **Save**.

Appendix A List of Acronyms

AD	Active Directory
ANC	Adaptive Network Control
API	Application Programming Interface
BIOS	Basic Input/Output System
CA	Certificate Authority
CLI	Command Line Interface
CPU	Central Processing Unit
CSV	Comma-Separated Values
CVSS	Common Vulnerability Scoring System
DEP	Device Enrollment Program
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
EPEL	Extra Packages for Enterprise Linux
EULA	End User License Agreement
FMC	(Cisco) Firepower Management Center
FQDN	Fully Qualified Domain Name
FTD	(Cisco) Firepower Threat Defense
GB	Gigabyte
GUI	Graphical User Interface
HDD	Hard Disk Drive
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
ISE	(Cisco) Identity Services Engine
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MAC	Media Access Control
MDM	Mobile Device Management
MES	(Lookout) Mobile Endpoint Security
MNT	Monitoring
MSI	(Microsoft) Windows Installer

NAT	Network Address Translation
NCCoE	National Cybersecurity Center of Excellence
NIST	National Institute of Standards and Technology
OS	Operating System
OVA	Open Virtualization Appliance
OVF	Open Virtualization Format
PCI	Peripheral Component Interconnect
RaaS	Returner as a Service
RADIUS	Remote Authentication Dial-In User Service
RAM	Random-Access Memory
REST	Representational State Transfer
RPM	RPM Package Manager
SaaS	Software as a Service
SCCM	(Microsoft) System Center Configuration Manager
SGT	Security Group Tag
SMBIOS	System Management Basic Input/Output System
SMS	(Microsoft) Systems Management Server
SNMP	Simple Network Management Protocol
SP	Special Publication
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
TLS	Transport Layer Security
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network
VPR	Vulnerability Prioritization Rating
WAN	Wide Area Network
WSUS	Windows Server Update Services