Secure Configuration Guide

03 May 2018
vRealize Automation 7.3

vmware®

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

Contents

Secure Configuration

Secure Configuration helps users to evaluate and optimize the secure configuration of vRealize Automation deployments.

Secure Configuration describes verification and configuration of secure deployments for typical vRealize Automation environments and provides information and procedures to help users make informed choices regarding security configuration.

Intended Audience

This information is intended for vRealize Automation system administrators and other users who are responsible for system security maintenance and configuration.

VMware Technical Publications Glossary

VMware Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation, go to http://www.vmware.com/support/pubs.

Updated Information

2

This *Secure Configuration Guide* is updated with each release of the product or when necessary.

This table provides the update history of the *Secure Configuration Guide*.

Revision	Description
May 3, 2018	Minor edits.
December 5, 2017	Updated Enable TLS on Localhost Configuration
002535-01	Updated Set vRealize Automation appliance Session Timeout.
002535-00	Initial release.

vRealize Automation Secure Baseline Overview

3

VMware provides comprehensive recommendations to help you verify and configure a secure baseline for your vRealize Automation system.

Use the appropriate tools and procedures as specified by VMware to verify and maintain a secure, hardened baseline configuration for your vRealize Automation system. Some vRealize Automation components are installed in a hardened or partially-hardened state, but you should review and verify configuration of each component in light of VMware security recommendations, company security policies, and known threats.

vRealize Automation Security Posture

The security posture of vRealize Automation assumes a holistically secure environment based on system and network configuration, organizational security policies, and security best practices.

When verifying and configuring hardening of a vRealize Automation system, consider each of the following areas as addressed by VMware hardening recommendations.

- Secure Deployment
- Secure Configuration
- Network Security

To ensure that your system is securely hardened, consider VMware recommendations and your local security policies as they relate to each of these conceptual areas.

System Components

When considering hardening and the secure configuration of your vRealize Automation system, ensure that you understand all components and how they work together to support system functionality.

Consider the following components when planning and implementing a secure system.

- vRealize Automation appliance
- IaaS Component

To familiarize yourself with vRealize Automation and how the components operate together, see *Foundations and Concepts* in the VMware vRealize Automation documentation center. For information about typical vRealize Automation deployments and architecture, see *Reference Architecture*.

Verifying the Integrity of Installation Media

4

Users should always verify the integrity of the installation media before installing a VMware product.

Always verify the SHA1 hash after you download an ISO, offline bundle, or patch to ensure integrity and authenticity of the downloaded files. If you obtain physical media from VMware and the security seal is broken, return the software to VMware for a replacement.

After you download the media, use the MD5/SHA1 sum value to verify the integrity of the download. Compare the MD5/SHA1 hash output with the value posted on the VMware Web site. SHA1 or MD5 hash should match.

For more information about verifying the integrity of the installation media, see http://kb.vmware.com/kb/1537.

Hardening VMware System Software Infrastructure

5

As part of your hardening process, assess the deployed software infrastructure that supports your VMware system and verify that it meets VMware hardening guidelines.

Before hardening your VMware system, review and address security deficiencies in your supporting software infrastructure to create a completely hardened and secure environment. Software infrastructure elements to consider include operating system components, supporting software, and database software. Address security concerns in these and other components according to the manufacturer's recommendations and other relevant security protocols.

This chapter includes the following topics:

- Hardening the VMware vSphere ® Environment
- Hardening the Infrastructure as a Service Host
- Hardening Microsoft SQL Server
- Hardening Microsoft .NET
- Hardening Microsoft Internet Information Services (IIS)

Hardening the VMware vSphere ® Environment

Assess the VMware vSphere ® environment and verify that the appropriate level of vSphere hardening guidance is enforced and maintained.

For more guidance about hardening, see http://www.vmware.com/security/hardening-guides.html .

As part of a comprehensively hardened environment, VMware vSphere ® infrastructure must meet security guidelines as defined by VMware.

Hardening the Infrastructure as a Service Host

Verify that your Infrastructure as a Service Microsoft Windows host machine is hardened according to VMware guidelines.

Review the recommendations in the appropriate Microsoft Windows hardening and secure best practice guidelines, and ensure that your Windows Server host is appropriately hardened. Not following the hardening recommendations might result in exposure to known security vulnerabilities from insecure components on Windows releases.

To verify that your version is supported, see the vRealize Automation Support Matrix.

Contact your Microsoft vendor about the correct guidance for hardening practices of Microsoft products.

Hardening Microsoft SQL Server

Verify that the Microsoft SQL Server database meets security guidelines as established by Microsoft and VMware.

Review the recommendations in the appropriate Microsoft SQL Server hardening and secure best practice guidelines. Review all Microsoft security bulletins regarding the installed version of Microsoft SQL Server. Not following the hardening recommendations might result in exposure to known security vulnerabilities from insecure components on Microsoft SQL Server versions.

To verify that your version Microsoft SQL Server is supported, see the vRealize Automation Support Matrix.

Contact your Microsoft vendor for guidance about hardening practices for Microsoft products.

Hardening Microsoft .NET

As part of a comprehensively hardened environment, Microsoft .NET must meet security guidelines as laid out by Microsoft and VMware.

Review the recommendations set out in the appropriate .NET hardening and secure best practice guidelines. Also, review all Microsoft security bulletins regarding the version of Microsoft SQL Server you are using. Failure to follow the hardening recommendations might result in exposure to known security vulnerabilities from insecure Microsoft.NET components.

To verify that your version of Microsoft.NET is supported, see the vRealize Automation Support Matrix.

Contact your Microsoft vendor for guidance on hardening practices for Microsoft products.

Hardening Microsoft Internet Information Services (IIS)

Verify that your Microsoft Internet Information Services (IIS) meet all Microsoft and VMware security guidelines.

Review the recommendations set out in the appropriate Microsoft IIS hardening and secure best practice guidelines. Also, review all Microsoft security bulletins regarding the version of IIS you are using. Not following the hardening recommendations might result in exposure to known security vulnerabilities.

To verify that your version is supported, see the vRealize Automation Support Matrix.

Contact your Microsoft vendor for guidance on hardening practices for Microsoft products.

Reviewing Installed Software

6

Because vulnerabilities in third party and unused software increase the risk of unauthorized system access and disruption of availability, it is important to review all software installed on VMware host machines and evaluate its use.

Do not install software that is not required for the secure operation of the system on the VMware host machines. Uninstall unused or extraneous software.

Inventory Installed Unsupported Software

Assess your VMware deployment and inventory of installed products to verify that no extraneous unsupported software is installed.

For more information about the support policies for third-party products, see the VMware support article at https://www.vmware.com/support/policies/thirdparty.html.

Verify Third-Party Software

VMware does not support or recommend installation of third party software that has not been tested and verified. Insecure, unpatched, or unauthenticated third-party software installed on VMware host machines might put the system at risk of unauthorized access and disruption of availability. If you must use unsupported third-party software, consult the third-party vendor for secure configuration and patching requirements.

VMware Security Advisories and Patches

7

To maintain maximum security for your system, follow VMware security advisories and apply all relevant patches.

VMware releases security advisories for products. Monitor these advisories to ensure that your product is protected against known threats.

Assess the vRealize Automation installation, patching, and upgrade history and verify that the released VMware Security Advisories are followed and enforced.

For more information about the current VMware security advisories, see http://www.vmware.com/security/advisories/.

Secure Configuration

8

Verify and update security settings for vRealize Automation virtual appliances and the Infrastructure as a Service component as appropriate for your system configuration. In addition, verify and update configuration of other components and applications.

Securely configuring a vRealize Automation installation involves addressing the configuration of each component individually and as they work together. Consider the configuration of all systems components in concert to achieve a reasonably secure baseline.

This chapter includes the following topics:

- Securing the vRealize Automation Appliance
- Securing the Infrastructure as a Service Component

Securing the vRealize Automation Appliance

Verify and update security settings for the vRealize Automation appliance as necessary for your system configuration.

Configure security settings for your virtual appliances and their host operating systems. In addition, set or verify configuration of other related components and applications. In some cases, you need to verify existing settings, while in others you must change or add settings to achieve an appropriate configuration.

Change the Root Password

You can change the root password for the vRealize Automation appliance to meet applicable security requirements.

Change the root password on the vRealize Automation appliance using the Virtual Appliance Management Interface. Verify that the root password meets your organization's corporate password complexity requirements.

Procedure

1   Open the Virtual Appliance Management Interface for your vRealize Automation appliance.

     https://*vRealizeAppliance-url*:5480

2   Select the **Admin** tab on the Virtual Appliance Management Interface.

3   Select the **Admin** submenu.

4 Enter the existing password in the **Current administrator password** text box.

5 Enter the new password in the **New administrator password** text box.

6 Enter the new password in the **Retype new administrator password** text box.

7 Click **Save Settings** to save your changes.

Verify Root Password Hash and Complexity

Verify that the root password meets your organization's corporate password complexity requirements.

Validating the root password complexity is required as the root user bypasses the pam_cracklib module password complexity check that is applied to user accounts.

The account password must start with 6, which indicates a sha512 hash. This is the standard hash for all hardened appliances.

Procedure

1 To verify the hash of the root password, log in as root and run the `# more /etc/shadow` command.

The hash information is displayed.

Figure 8-1. Password Hash Results


```
vcac148-084-111:~ # more /etc/shadow
bin:*:16332:0:60:7:::
daemon:*:16332:0:60:7:::
haldaemon:*:16332:0:60:7:::
mail:*:15870::60::::
man:*:16332:0:60:7:::
messagebus:*:16332:0:60:7:::
nobody:*:15870::60::::
ntp:*:16332:0:60:7:::
polkituser:*:16332:0:60:7:::
postfix:*:16332:0:60:7:::
root:$6$PHxGPy5A$ba8KezK4SS44UEHPfAtgs
P/:16346:0:365:7:::
```

2 If the root password does not contain a sha512 hash, run the `passwd` command to change it.

All hardened appliances enable `enforce_for_root` for the `pw_history` module, found in the `/etc/pam.d/common-password` file. The system remembers the last five passwords by default. Old passwords are stored for each user in the `/etc/securetty/passwd` file.

Verify Root Password History

Verify that the password history is enforced for the root account.

All hardened appliances enable `enforce_for_root` for the pw_history module, found in the `/etc/pam.d/common-password` file. The system remembers the last five passwords by default. Old passwords are stored for each user in the `/etc/securetty/passwd` file.

Procedure

1   Run the following command:

    cat /etc/pam.d/common-password-vmware.local | grep pam_pwhistory.so

2   Ensure that `enforce_for_root` appears in the returned results.

    password required pam_pwhistory.so enforce_for_root remember=5 retry=3

Manage Password Expiry

Configure all account password expirations in accordance with your organization's security policies.

By default, all hardened VMware virtual appliance accounts use a 60-day password expiration. On most hardened appliances, the root account is set to a 365-day password expiration. As a best practice, verify that the expiration on all accounts meets both security and operation requirements standards.

If the root password expires, you cannot reinstate it. You must implement site-specific policies to prevent administrative and root passwords from expiring.

Procedure

1   Log in to your virtual appliance machines as root and run the following command to verify the password expiration on all accounts.

    # cat /etc/shadow

    The password expiration is the fifth field (fields are separated by colons) of the shadow file. The root expiration is set in days.

    Figure 8-2. Password Expiry Field
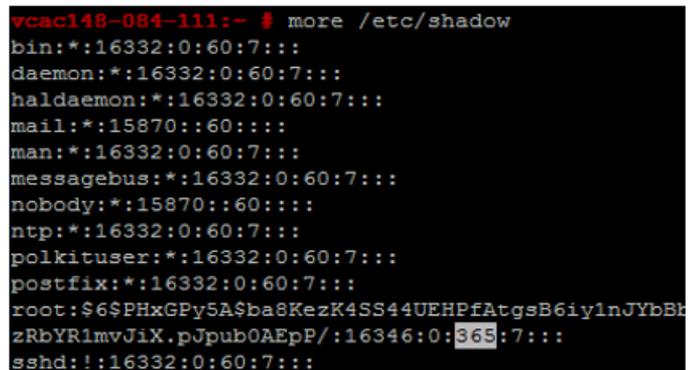
2   To modify the expiry of the root account, run a command of the following form.

    # passwd -x 365 root

    In this command, 365 specifies the number of days until password expiry. Use the same command to modify any user, substituting the specific account for 'root', and replacing the number of days to meet the expiry standards of the organization.

Managing Secure Shell and Administrative Accounts

For remote connections, all hardened appliances include the Secure Shell (SSH) protocol. Use SSH only when necessary and manage it appropriately to preserve system security.

SSH is an interactive command-line environment that supports remote connections to VMware virtual appliances. By default, SSH access requires high-privileged user account credentials. Root user SSH activities generally bypass the role-based access control (RBAC) and audit controls of the virtual appliances.

As a best practice, disable SSH in a production environment, and activate it only to troubleshoot problems that you cannot resolve by other means. Leave it enabled only while needed for a specific purpose and in accordance with your organization's security policies. SSH is disabled by default on the vRealize Automation appliance. Depending on your vSphere configuration, you might enable or disable SSH when you deploy your Open Virtualization Format (OVF) template.

As a simple test to determine whether SSH is enabled on a machine, try opening a connection by using SSH. If the connection opens and requests credentials, then SSH is enabled and available for connections.

Secure Shell root User Account

Because VMware appliances do not include pre-configured user accounts, the root account can use SSH to directly log in by default. Disable SSH as root as soon as possible.

To meet the compliance standards for non repudiation, the SSH server on all hardened appliances is pre-configured with the AllowGroups wheel entry to restrict SSH access to the secondary group wheel. For separation of duties, you can modify the AllowGroups wheel entry in the `/etc/ssh/sshd_config` file to use another group such as sshd.

The wheel group is enabled with the `pam_wheel` module for superuser access, so members of the wheel group can su-root, where the root password is required. Group separation enables users to SSH to the appliance, but not to su to root. Do not remove or modify other entries in the AllowGroups field, which ensures proper appliance functionality. After making a change, you must restart the SSH daemon by running the command: `# service sshd restart`.

Enable or Disable Secure Shell on the vRealize Automation Appliances

Enable Secure Shell (SSH) on the vRealize Automation appliance only for troubleshooting. Disable SSH on these components during normal production operation.

You can enable or disable SSH on the vRealize Automation appliance using the Virtual Appliance Management console.

Procedure

1   Navigate to the Virtual Appliance Management Console (VAMI) for your vRealize Automation appliance.

    : https://*vRealizeAppliance url*:5480

2   Click the **Admin** tab.

3   Click the **Admin** sub-menu.

4   Select the **SSH service enable** check box to enable SSH or deselect it to disable SSH.

5   Click **Save Settings** to save your changes.

Create Local Administrator Account for Secure Shell

As a security best practice, create and configure local administrative accounts for Secure Shell (SSH) on your virtual appliance host machines. Also, remove root SSH access after you create the appropriate accounts.

Create local administrative accounts for SSH, or members of the secondary wheel group, or both. Before you disable direct root access, test that authorized administrators can access SSH by using AllowGroups, and that they can su to root using the wheel group.

Procedure

1   Log in to the virtual appliance as root and run the following commands with the appropriate username.

```
# useradd –g users <username> –G wheel –m –d /home/username
            # passwd username
```

Wheel is the group specified in AllowGroups for ssh access. To add multiple secondary groups, use `–G wheel,sshd`.

2   Switch to the user and provide a new password to enforce password complexity checking.

```
# su –username
    # username@hostname:~>passwd
```

If the password complexity is met, the password updates. If the password complexity is not met, the password reverts to the original password, and you must rerun the password command.

3   To remove direct login to SSH, modify the`/etc/ssh/sshd_config` file by replacing `(#)PermitRootLogin yes` with `PermitRootLogin no`.

Alternatively, you can enable/disable SSH in the Virtual Appliance Management Interface (VAMI) by selecting or deselecting the **Administrator SSH login enabled** check box on the **Admin** tab.

What to do next

Disable direct logins as root. By default, the hardened appliances allow direct login to root through the console. After you create administrative accounts for non-repudiation and test them for su-root wheel access, disable direct root logins by editing the `/etc/security` file as root and replacing the `tty1` entry with `console`.

1 Open the `/etc/securetty` file in a text editor.

2 Locate `tty1` and replace it with `console`.

3 Save the file and close it.

Harden the Secure Shell Server Configuration

Where possible, all VMware appliances have a default hardened configuration. Users can verify that their configuration is appropriately hardened by examining the server and client service settings in the global options section of the configuration file.

Procedure

1 Open the `/etc/ssh/sshd_config` server configuration file on the VMware appliance, and verify that the settings are correct.

Setting	Status
Server Daemon Protocol	Protocol 2
CBC Ciphers	aes256-ctr and aes128-ctr
TCP Forwarding	AllowTCPForwarding no
Server Gateway Ports	Gateway Ports no
X11 Forwarding	X11Forwarding no
SSH Service	Use the AllowGroups field and specify a group permitted access. Add appropriate members to this group.
GSSAPI Authentication	GSSAPIAuthentication no, if unused
Keberos Authentication	KeberosAuthentication no, if unused
Local Variables (AcceptEnv global option)	Set to `disabled by commenting out` or `enabled for LC_* or LANG variables`
Tunnel Configuration	PermitTunnel no
Network Sessions	MaxSessions 1
User Concurrent Connections	Set to 1 for root and any other user. The `/etc/security/limits.conf` file also needs to be configured with the same setting.
Strict Mode Checking	Strict Modes yes
Privilege Separation	UsePrivilegeSeparation yes
rhosts RSA Authentication	RhostsESAAuthentication no
Compression	Compression delayed or Compression no

Setting	Status
Message Authentication code	MACs hmac-sha1
User Access Restriction	PermitUserEnvironment no

2 Save your changes and close the file.

Harden the Secure Shell Client Configuration

As part of your system hardening process, verify hardening of the SSH client by examining the SSH client configuration file on virtual appliance host machines to ensure that it is configured according toVMware guidelines.

Procedure

1 Open the SSH client configuration file, `/etc/ssh/ssh_config`, and verify that settings in the global options section are correct.

Setting	Status
Client Protocol	Protocol 2
Client Gateway Ports	Gateway Ports no
GSSAPI Authentication	GSSAPIAuthentication no
Local Variables (SendEnv global option)	Provide only LC_* or LANG variables
CBC Ciphers	aes256-ctr and aes128-ctr only
Message Authentication Codes	Used in the MACs `hmac-sha1` entry only

2 Save your changes and close the file.

Verifying Secure Shell Key File Permissions

To minimize the possibility of malicious attacks, maintain critical SSH key file permissions on your virtual appliance host machines.

After configuring or updating your SSH configuration, always verify that the following SSH key file permissions do not change.

- The public host key files located in `/etc/ssh/*key.pub` are owned by the root user and have permissions set to 0644 (-rw-r--r--).

- The private host key files located in `/etc/ssh/*key` are owned by the root user and have permissions set to 0600 (-rw------).

Verify SSH Key File Permissions

Verify that SSH permissions are applied to both public and private key files.

Procedure

1 Check the SSH public key files by running the following command: `ls –l /etc/ssh/*key.pub`

2   Verify that the owner is root, that the group owner is root, and that the files have permissions set to 0644 (-rw-r--r--).

3   Fix any problems by running the following commands.

    chown root /etc/ssh/*key.pub

    chgrp root /etc/ssh/*key.pub

    chmod 644 /etc/ssh/*key.pub

4   Check the SSH private key files by running the following command: `ls –l /etc/ssh/*key`

5   Fix any problems by running the following commands.

    chown root /etc/ssh/*key

    chgrp root /etc/ssh/*key

    chmod 644 /etc/ssh/*key

Change the Virtual Appliance Management Interface User

You can add and delete users on the Virtual Appliance Management Interface to create the appropriate level of security.

The root user account for the Virtual Appliance Management Interface uses PAM for authentication, so the clipping levels set by PAM also apply. If you have not appropriately isolated the Virtual Appliance Management Interface, a lock out of the system root account could occur if an attacker attempts to brute force the login. In addition, where the root account is considered insufficient to provide non-repudiation by more than one person in your organization, then you might elect to change the admin user for the management interface.

Prerequisites

Procedure

1   Run the following command to create a new user and add it to the Virtual Appliance Management Interface group.

    useradd –G vami,root *user*

2   Create a password for the user.

    passwd *user*

3   (Optional) Run the following command to disable root access on the Virtual Appliance Management Interface.

    usermod –R vami root

    Note   Disabling root access to the Virtual Appliance Management Interface also disables the ability to update the Administrator, or root, password from the Admin tab.

Set Boot Loader Authentication

To provide an appropriate level of security, configure boot loader authentication on your VMware virtual appliances.

If the system's boot loader requires no authentication, users with system console access can alter the system boot configuration or boot the system into single user or maintenance mode, which can result in denial of service or unauthorized system access. Because boot loader authentication is not set by default on the VMware virtual appliances, you must create a GRUB password to configure it.

Procedure

1   Verify whether a boot password exists by locating the `password --md5 <password-hash>` line in the `/boot/grub/menu.lst` file on your virtual appliances.

2   If no password exists, run the `# /usr/sbin/grub-md5-crypt` command on your virtual appliance.

    An MD5 password is generated, and the command supplies the md5 hash output.

3   Append the password to the `menu.lst` file by running the `# password --md5 <hash from grub-md5-crypt>` command.

Configure NTP

For critical time sourcing, disable host time synchronization and use the Network Time Protocol (NTP) on the vRealize Automation appliance.

The NTP daemon on vRealize Automation appliance provides synchronized time services. NTP is disabled by default, so you need to configure it manually. If possible, use NTP in production environments to track user actions and to detect potential malicious attacks and intrusions through accurate audit and log keeping. For information about NTP security notices, see the NTP Web site.

The NTP configuration file is located in the `/etc/` folder on each appliance. You can enable the NTP service for the vRealize Automation appliance and add time servers on the **Admin** tab of the Virtual Appliance Management Interface.

Procedure

1   Open the `/etc/ntp.conf` configuration file on your virtual appliance host machine using a text editor.

2   Set the file ownership to **root:root**.

3   Set the permissions to **0640**.

4   To mitigate the risk of a denial-of-service amplification attack on the NTP service, open the `/etc/ntp.conf` file and ensure that the restrict lines appear in the file.

```
restrict default kod nomodify notrap nopeer noquery
restrict -6 default kod nomodify notrap nopeer noquery
restrict 127.0.0.1
restrict -6 ::1
```

5   Save any changes and close the files.

Configuring TLS for vRealize Automation Appliance Data In-transit

Ensure that your vRealize Automation deployment uses strong TLS protocols to secure transmission channels for vRealize Automation appliance components.

For performance considerations, TLS is not enabled for localhost connections between some application services. Where defence in depth is of concern, enable TLS on all localhost communications.

Important   If you are terminating TLS on the load balancer, disable insecure protocols such as SSLv2, SSLv3, and TLS 1.0 on all load balancers.

Enable TLS on Localhost Configuration

By default some localhost communication does not use TLS. You can enable TLS across all localhost connections to provide enhanced security.

Procedure

1   Connect to the vRealize Automation appliance using SSH.

2   Set permissions for the vcac keystore by running the following commands.

```
usermod —A vco,coredump,pivotal vco
chown vcac.pivotal /etc/vcac/vcac.keystore
chmod 640 /etc/vcac/vcac.keystore
```

3   Update the HAProxy configuration.

   a   Locate the lines containing the following string

   `server local 127.0.0.1`… and add the following to the end of such lines: `ssl verify none`

   This section contains other lines like the following:

backend-horizon	backend-vro
backend-vra	backend-artifactory
backend-vra-health	

   b   Change the port for backend-horizon from 8080 to 8443.

4   Get the password of keystorePass.

    a   Locate the property `certificate.store.password` in the `/etc/vcac/security.properties` file.

        For example, `certificate.store.password=s2enc~iom0GXATG+RB8ff7Wdm4Bg==`

    b   Decrypt the value using the following command:

        `vcac-config prop-util -d --p VALUE`

        For example, `vcac-config prop-util -d --p s2enc~iom0GXATG+RB8ff7Wdm4Bg==`

5   Configure the vRealize Automation service

    a   Open the `/etc/vcac/server.xml` file.

    b   Add the following attribute to the Connector tag, replacing certificate.store.password with the certificate store password value found in `etc/vcac/security.properties`.

```
scheme="https" secure="true" SSLEnabled="true" sslProtocol="TLS"
keystoreFile="/etc/vcac/vcac.keystore" keyAlias="apache"
keystorePass="certificate.store.password"
```

6   Configure the vRealize Orchestrator service.

    a   Open the `/etc/vco/app/server.xml` file

    b   Add the following attribute to the Connector tag, replacing certificate.store.password with the certificate store password value found in `etc/vcac/security.properties`.

```
scheme="https" secure="true" SSLEnabled="true" sslProtocol="TLS"
keystoreFile="/etc/vcac/vcac.keystore" keyAlias="apache"
keystorePass="certificate.store.password"
```

7   Restart the vRealize Orchestrator, vRealize Automation, and haproxy services.

```
service vcac-server restart
service vco-server restart
service haproxy restart
```

Note   If the vco-server does not restart, reboot the host computer.

8   Configure the Virtual Appliance Management Interface.

    a   Open the `/opt/vmware/share/htdocs/service/café-services/services.py` file.

    b   Change the `conn = httplib.HTTP()` line to `conn = httplib.HTTPS()` to enhance security.

Enable Federal Information Processing Standard (FIPS) 140-2 Compliance

The vRealize Automation appliance now uses the Federal Information Processing Standard (FIPS) 140-2 certified version of OpenSSL for data-in-transit over TLS on all inbound and outbound network traffic.

You can enable or disable FIPS mode in the vRealize Automation appliance management interface. You can also configure FIPS from the command line while logged in as root, using the following commands:

```
vcac-vami fips enable
vcac-vami fips disable
vcac-vami fips status
```

When FIPS is enabled, inbound and outbound vRealize Automation appliance network traffic on port 443 uses FIPS 140–2 compliant encryption. Regardless of the FIPS setting, vRealize Automation uses AES–256 to protect secured data stored on the vRealize Automation appliance.

Note Currently vRealize Automation only partially enables FIPS compliance, because some internal components do not yet use certified cryptographic modules. In cases where certified modules have not yet been implemented, the AES–256 based encryption is used in all cryptographic algorithms.

Note The following procedure will reboot the physical machine when you alter the configuration.

Procedure

1 Log in as root to the vRealize Automation appliance management interface.

   `https:// vrealize-automation-appliance-FQDN:5480`

2 Select **vRA Settings > Host Settings**.

3 Click the button under the Actions heading on the upper right to enable or disable FIPS.

4 Click **Yes** to restart the vRealize Automation appliance

Verify that SSLv3, TLS 1.0, and TLS 1.1 are Disabled

As part of your hardening process, ensure that the deployed vRealize Automation appliance uses secure transmission channels.

Prerequisites

Complete Enable TLS on Localhost Configuration.

Procedure

1   Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled in the HAProxy https handlers on the vRealize
    Automation appliance.

Review this file	Ensure the following is present	In the appropriate line as shown
/etc/haproxy/conf.d/20-vcac.cfg	no-sslv3 no-tlsv10 no-tls11 force-tls12	bind 0.0.0.0:443 ssl crt /etc/apache2/server.pem ciphers !aNULL:! eNULL:kECDH+AESGCM:ECDH+AESGCM:RSA +AESGCM:kECDH+AES:ECDH+AES:RSA+AES:@STRENGTH no-sslv3 no-tlsv10 no-tlsv11
/etc/haproxy/conf.d/30-vro-config.cfg	no-sslv3 no-tlsv10 no-tls11 force-tls12	bind :::8283 v4v6 ssl crt /opt/vmware/etc/lighttpd/server.pem ciphers !aNULL:!eNULL:kECDH+AESGCM:ECDH +AESGCM:RSA+AESGCM:kECDH+AES:ECDH+AES:RSA +AES:@STRENGTH no-sslv3 no-tlsv10 no-tlsv11

2   Restart the service.

```
service haproxy restart
```

3   Open the /opt/vmware/etc/lighttpd/lighttpd.conf file, and verify that the correct disable
    entries appear.

    Note   There is no directive to disable TLS 1.0 or TLS 1.1 in Lighttpd. The restriction on TLS 1.0 and
    TLS 1.1 use can be partially mitigated by enforcing OpenSSL to not use cipher suites of TLS 1.0 and
    TLS 1.1.

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
```

4   Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled for the Console Proxy on the vRealize
    Automation appliance.

    a   Edit the /etc/vcac/security.properties file by adding or modifying the following line:

    ```
    consoleproxy.ssl.server.protocols = TLSv1.2
    ```

    b   Restart the server by running the following command:

    ```
    service vcac-server restart
    ```

5 Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled for the vCO service.

    a Locate the `<Connector>` tag in the `/etc/vco/app-server/server.xml` file and add the following attribute:

```
sslEnabledProtocols = "TLSv1.2"
```

    b Restart the vCO service by running the following command.

```
service vco-server restart
```

6 Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled for the vRealize Automation service.

    a Add the following attributes to the `<Connector>` tag in the `/etc/vcac/server.xml` file

```
sslEnabledProtocols = "TLSv1.2"
```

    b Restart the vRealize Automation service by running the following command:

```
service vcac-server restart
```

7 Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled for RabbitMQ.

Open the `/etc/rabbitmq/rabbitmq.config` file and verify that `{versions, ['tlsv1.2', 'tlsv1.1']}` are present in the ssl and ssl_options sections.

```
[
  {ssl, [
      {versions, ['tlsv1.2', 'tlsv1.1']},
      {ciphers, ["AES256-SHA", "AES128-SHA"]}
  ]},
   {rabbit, [
      {tcp_listeners, [{"127.0.0.1", 5672}]},
      {frame_max, 262144},
      {ssl_listeners, [5671]},
      {ssl_options, [
         {cacertfile, "/etc/rabbitmq/certs/ca/cacert.pem"},
         {certfile, "/etc/rabbitmq/certs/server/cert.pem"},
         {keyfile, "/etc/rabbitmq/certs/server/key.pem"},
         {versions, ['tlsv1.2', 'tlsv1.1']},
         {ciphers, ["AES256-SHA", "AES128-SHA"]},
         {verify, verify_peer},
         {fail_if_no_peer_cert, false}
      ]},
      {mnesia_table_loading_timeout,600000},
      {cluster_partition_handling, autoheal},
      {heartbeat, 600}
   ]},
   {kernel, [{net_ticktime,  120}]}
 ].
```

8 Restart the RabbitMQ server.

```
# service rabbitmq-server restart
```

9   Verify that SSLv3, TLS 1.0, and TLS 1.1 are disabled for the vIDM service.

Open the `opt/vmware/horizon/workspace/conf/server.xml` file for each instance of the connector containing `SSLEnabled="true"` and ensure that the following line is present.

`sslEnabledProtocols="TLSv1.2"`

Disable TLS 1.0

Disable TLS 1.0 in applicable vRealize Automation components.

There is no directive to disable TLS 1.0 in Lighttpd. The restriction on TLS 1.0 use can be partially mitigated by enforcing that OpenSSL does not use cipher suites of TLS 1.0 as described in step 2 below.

Procedure

1   Disable TLS 1.0 in the HAProxy https handler on the vRealize Automation appliance.

    a   Append `no-tlsv10` to the end of the following entry in the `/etc/haproxy/conf.d/20-vcac.cfg` file.

       `bind 0.0.0.0:443 ssl crt /etc/apache2/server.pem ciphers TLSv1+HIGH:!aNULL:!eNULL:!3DES:!RC4:!CAMELLIA:!DH:!kECDHE:@STRENGTH no-sslv3 no-tlsv10`

    b   Append no-tlsv10 to the end of the following entry in the `/etc/haproxy/conf.d/30-vro-config.cfg` file.

       `bind :8283 ssl crt /opt/vmware/etc/lighttpd/server.pem ciphers TLSv1+HIGH:!aNULL:!eNULL:!3DES:!RC4:!CAMELLIA:!DH:!kECDHE:@STRENGTH no-sslv3 no-tls10`

Note   To re-enable TLS 1.0 remove `no-tlsv10` from the bind directive.

---

2   Verify in Lighttpd that OpenSSL does not use cipher suites of TLS 1.0

    a   Edit the `ssl.cipher-list` line in the `/opt/vmware/etc/lighttpd/lighttpd.conf` file as follows.

```
ssl.cipher-list = "ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-
CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
RSA-AES128-SHA256"
```

    b   Restart lighttpd using the following command:

       `service vami-lighttp restart`

3 Disable TLS 1.0 for the Console Proxy on the vRealize Automation appliance.

    a   Add or modify the following line in the `/etc/vcac/security.properties` file.

```
consoleproxy.ssl.server.protocols = TLSv1.2, TLSv1.1
```

    b   Restart the server by running the following command:

```
service vcac-server restart
```

Note To re-enable TLS 1.0, add TLSv1 as follows and then restart the vcac-server service:

```
consoleproxy.ssl.server.protocols = TLSv1.2,TLSv1.1, TLSv1
```

4 Disable TLS 1.0 for the vCO service.

    a   Locate the `<Connector>` tag in the `/etc/vco/app-server/server.xml` file and add the following attribute to it:

```
sslEnabledProtocols = "TLSv1.1,TLSv1.2"
```

    b   Restart the vCO service by running the following command:

```
service vco-server restart
```

5 Disable TLS 1.0 for the vRealize Automation service.

    a   Locate `<Connector>` tag in the `/etc/vcac/server.xml` file and add the following attribute to it:

```
sslEnabledProtocols = "TLSv1.1,TLSv1.2"
```

    b   Restart the vRealize Automation service by running the following commands.

```
service vcac-server restart
```

Note To re-enable TLS 1.0, add TLSv1 to sslEnabledProtocols. For example,
`sslEnabledProtocols = "TLSv1.1,TLSv1.2,TLSv1"`

6 Disable TLS 1.0 for RabbitMQ.

    a   Open the `/etc/rabbitmq/rabbitmq.config` file and verify that tlsv1.2 and tlsv1.1 are added to the ssl and ssl_options sections as shown in the following example.

```
[
   {ssl, [
      {versions, ['tlsv1.2', 'tlsv1.1']},
      {ciphers, ["AES256-SHA", "AES128-SHA"]}
   ]},
   {rabbit, [
      {tcp_listeners, [{"127.0.0.1", 5672}]},
      {frame_max, 262144},
      {ssl_listeners, [5671]},
      {ssl_options, [
         {cacertfile, "/etc/rabbitmq/certs/ca/cacert.pem"},
         {certfile, "/etc/rabbitmq/certs/server/cert.pem"},
         {keyfile, "/etc/rabbitmq/certs/server/key.pem"},
         {versions, ['tlsv1.2', 'tlsv1.1']},
         {ciphers, ["AES256-SHA", "AES128-SHA"]},
         {verify, verify_peer},
         {fail_if_no_peer_cert, false}
      ]},
      {mnesia_table_loading_timeout,600000},
      {cluster_partition_handling, autoheal},
      {heartbeat, 600}
   ]},
   {kernel, [{net_ticktime,  120}]}
].
```

    b   Restart the RabbitMQ server by running the following command:

```
# service rabbitmq-server restart
```

Configuring TLS Cipher Suites for vRealize Automation Components

For maximum security, you must configure vRealize Automation components to use strong ciphers.

The encryption cipher negotiated between the server and the browser determines the encryption strength that is used in a TLS session.

To ensure that only strong ciphers are selected, disable weak ciphers in vRealize Automation components. Configure the server to support only strong ciphers and to use sufficiently large key sizes. Also, configure all ciphers in a suitable order.

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites. Also ensure that cipher suites using Diffie-Hellman (DHE) key exchange are disabled

Disable Weak Ciphers in HA Proxy

Review the vRealize Automation appliance HA Proxy Service ciphers against the list of acceptable ciphers and disable all of those considered weak.

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites.

Procedure

1 Review the `/etc/haproxy/conf.d/20-vcac.cfg` file ciphers entry of the bind directive and disable any that are considered weak.

   `bind 0.0.0.0:443 ssl crt /etc/apache2/server.pem ciphers !aNULL:!eNULL:kECDH`
   `+AESGCM:ECDH+AESGCM:RSA+AESGCM:kECDH+AES:ECDH+AES:RSA+AES:@STRENGTH no-sslv3 no-`
   `tlsv10 no-tlsv11`

2 Review the `/etc/haproxy/conf.d/30-vro-config.cfg` file ciphers entry of the bind directive and disable any that are considered weak.

   `bind :::8283 v4v6 ssl crt /opt/vmware/etc/lighttpd/server.pem ciphers !aNULL:!`
   `eNULL:kECDH+AESGCM:ECDH+AESGCM:RSA+AESGCM:kECDH+AES:ECDH+AES:RSA+AES:@STRENGTH`
   `no-sslv3 no-tlsv10 no-tlsv11`

Disable Weak Ciphers in the vRealize Automation appliancevRealize Automation Appliance Console Proxy Service

Review the vRealize Automation appliance Console Proxy Service ciphers against the list of acceptable ciphers and disable all of those considered weak.

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites.

Procedure

1 Open the `/etc/vcac/security.properties` file in a text editor.

2 Add a line to the file to disable the unwanted cipher suites.

   Use a variation of the following line:

   ```
   consoleproxy.ssl.ciphers.disallowed=cipher_suite_1, cipher_suite_2,etc
   ```

   For example, to disable the AES 128 and AES 256 cipher suites, add the following line:

   ```
   consoleproxy.ssl.ciphers.disallowed=TLS_DH_DSS_WITH_AES_128_CBC_SHA,
   TLS_DH_DSS_WITH_AES_256_CBC_SHA, TLS_DH_RSA_WITH_AES_256_CBC_SHA,
   TLS_DHE_DSS_WITH_AES_128_CBC_SHA, TLS_DHE_DSS_WITH_AES_256_CBC_SHA,
   TLS_DHE_RSA_WITH_AES_128_CBC_SHA, TLS_DHE_RSA_WITH_AES_256_CBC_SHA
   ```

3 Restart the server using the following command.

```
service vcac-server restart
```

Disable Weak Ciphers in vRealize Automation appliance vCO Service

Review vRealize Automation appliance vCO Service ciphers against the list of acceptable ciphers and disable all of those considered weak.

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites.

Procedure

1 Locate the <Connector> tag in /etc/vco/app/server/server.xml file.

2 Edit or add the cipher attribute to use the desired cipher suites.

Refer to the following example:

```
ciphers="TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_EC
DSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_GCM_
SHA384,TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,TLS_ECDHE_RSA_
WITH_AES_256_CBC_SHA384"
```

Disable Weak Ciphers in the vRealize Automation appliance RabbitMQ Service

Review vRealize Automation appliance RabbitMQ Service ciphers against the list of acceptable ciphers and disable all of those that are considered weak.

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites.

Procedure

1 Evaluate the supported cipher suites. by running the # /usr/sbin/rabbitmqctl eval 'ssl:cipher_suites().' command.

The ciphers returned in the following example represent only the supported ciphers. The RabbitMQ server does not use or advertise these ciphers unless configured to do so in the rabbitmq.config file.

```
["ECDHE-ECDSA-AES256-GCM-SHA384","ECDHE-RSA-AES256-GCM-SHA384",
 "ECDHE-ECDSA-AES256-SHA384","ECDHE-RSA-AES256-SHA384",
 "ECDH-ECDSA-AES256-GCM-SHA384","ECDH-RSA-AES256-GCM-SHA384",
 "ECDH-ECDSA-AES256-SHA384","ECDH-RSA-AES256-SHA384",
 "DHE-RSA-AES256-GCM-SHA384","DHE-DSS-AES256-GCM-SHA384",
 "DHE-RSA-AES256-SHA256","DHE-DSS-AES256-SHA256","AES256-GCM-SHA384",
 "AES256-SHA256","ECDHE-ECDSA-AES128-GCM-SHA256",
 "ECDHE-RSA-AES128-GCM-SHA256","ECDHE-ECDSA-AES128-SHA256",
```

```
"ECDHE-RSA-AES128-SHA256","ECDH-ECDSA-AES128-GCM-SHA256",
"ECDH-RSA-AES128-GCM-SHA256","ECDH-ECDSA-AES128-SHA256",
"ECDH-RSA-AES128-SHA256","DHE-RSA-AES128-GCM-SHA256",
"DHE-DSS-AES128-GCM-SHA256","DHE-RSA-AES128-SHA256","DHE-DSS-AES128-SHA256",
"AES128-GCM-SHA256","AES128-SHA256","ECDHE-ECDSA-AES256-SHA",
"ECDHE-RSA-AES256-SHA","DHE-RSA-AES256-SHA","DHE-DSS-AES256-SHA",
"ECDH-ECDSA-AES256-SHA","ECDH-RSA-AES256-SHA","AES256-SHA",
"ECDHE-ECDSA-DES-CBC3-SHA","ECDHE-RSA-DES-CBC3-SHA","EDH-RSA-DES-CBC3-SHA",
"EDH-DSS-DES-CBC3-SHA","ECDH-ECDSA-DES-CBC3-SHA","ECDH-RSA-DES-CBC3-SHA",
"DES-CBC3-SHA","ECDHE-ECDSA-AES128-SHA","ECDHE-RSA-AES128-SHA",
"DHE-RSA-AES128-SHA","DHE-DSS-AES128-SHA","ECDH-ECDSA-AES128-SHA",
"ECDH-RSA-AES128-SHA","AES128-SHA"]
```

2   Select supported ciphers that meet the security requirements for your organization.

For example, to allow only `ECDHE-ECDSA-AES128-GCM-SHA256` & `ECDHE-ECDSA-AES256-GCM-SHA384`, review the `/etc/rabbitmq/rabbitmq.config` file and add the following line to ssl and ssl_options.

`{ciphers, ["ECDHE-ECDSA-AES128-GCM-SHA256", "ECDHE-ECDSA-AES256-GCM-SHA384"]}`

3   Restart the RabbitMQ server using the following command.

`service rabbitmq-server restart`

Verifying Security of Data-at-Rest

Verify the security of database users and accounts used with vRealize Automation.

Postgres User

The postgres linux user account is tied to the postgres database superuser account role, by default it is a locked account. This is the most secure configuration for this user as it is only accessible from the root user account. Do not unlock this user account.

Database User Account Roles

The default postgres user account roles should not be utilised for uses outside of application functionality. In order to support non-default database review or reporting activities, an additional account should be created and password appropriately protected.

Run the following script in the command line:

`vcac-vami add-db-user newUsername newPassword`

This will add a new user and a password provided by the user.

Note   This script must be ran against the master postgres database in the cases when master-slave HA postgres setup is configured.

Configure PostgreSQL Client Authentication

Ensure that local trust authentication, is not configured the vRealize Automation appliance PostgreSQL database. This configuration allows any local user, including the database super user, to connect as any PostgreSQL user without a password.

Note The Postgres super user account should remain as local trust.

The md5 authentication method is recommended because it sends encrypted passwords.

The client authentication configuration settings reside in the `/storage/db/pgdata/pg_hba.conf` file.

```
# TYPE  DATABASE          USER               ADDRESS           METHOD

# "local" is for Unix domain socket connections only
local      all              postgres                          trust
# IPv4 local connections:
#host    all              all            127.0.0.1/32      md5
hostssl    all              all             127.0.0.1/32      md5
# IPv6 local connections:
#host    all              all            ::1/128           md5
hostssl    all              all             ::1/128          md5


# Allow remote connections for VCAC user.
#host    vcac             vcac              0.0.0.0/0          md5
hostssl    vcac              vcac             0.0.0.0/0          md5
hostssl    vcac              vcac             ::0/0          md5
# Allow remote connections for VCAC replication user.
#host    vcac             vcac_replication   0.0.0.0/0          md5
hostssl    vcac              vcac_replication   0.0.0.0/0          md5
hostssl    vcac              vcac_replication   ::0/0          md5
# Allow replication connections by a user with the replication privilege.
#host    replication      vcac_replication   0.0.0.0/0          md5
hostssl    replication      vcac_replication   0.0.0.0/0          md5
hostssl    replication      vcac_replication   ::0/0          md5
```

If you edit the `pg_hba.conf` file, you must restart the Postgres server by running the following commands before changes can take effect.

```
# cd /opt/vmware/vpostgres/9.2/bin
# su postgres
# ./pg_ctl restart -D /storage/db/pgdata/ -m fast
```

Configure vRealize Automation Application Resources

Review vRealize Automation application resources and restrict file permissions.

Procedure

1   Run the following command to verify that files with SUID and GUID bits set are well-defined.

```
find / -path /proc -prune -o -type f -perm +6000 -ls
```

The following list should appear.

```
2197357   24 -rwsr-xr-x  1 polkituser root       23176 Mar 31  2015 /usr/lib/PolicyKit/polkit-
set-default-helper
2197354   16 -rwxr-sr-x  1 root     polkituser   14856 Mar 31  2015 /usr/lib/PolicyKit/polkit-
read-auth-helper
2197353   12 -rwsr-x---  1 root     polkituser   10744 Mar 31  2015 /usr/lib/PolicyKit/polkit-
grant-helper-pam
2197352   20 -rwxr-sr-x  1 root     polkituser   19208 Mar 31  2015 /usr/lib/PolicyKit/polkit-
grant-helper
2197351   20 -rwxr-sr-x  1 root     polkituser   19008 Mar 31  2015 /usr/lib/PolicyKit/polkit-
explicit-grant-helper
2197356   24 -rwxr-sr-x  1 root     polkituser   23160 Mar 31  2015 /usr/lib/PolicyKit/polkit-
revoke-helper
2188203  460 -rws--x--x  1 root     root        465364 Apr 21 22:38 /usr/lib64/ssh/ssh-keysign
2138858   12 -rwxr-sr-x  1 root     tty          10680 May 10  2010 /usr/sbin/utempter
2142482  144 -rwsr-xr-x  1 root     root        142890 Sep 15  2015 /usr/bin/passwd
2142477  164 -rwsr-xr-x  1 root     shadow      161782 Sep 15  2015 /usr/bin/chage
2142467  156 -rwsr-xr-x  1 root     shadow      152850 Sep 15  2015 /usr/bin/chfn
1458298  364 -rwsr-xr-x  1 root     root        365787 Jul 22  2015 /usr/bin/sudo
2142481   64 -rwsr-xr-x  1 root     root         57776 Sep 15  2015 /usr/bin/newgrp
1458249   40 -rwsr-x---  1 root     trusted      40432 Mar 18  2015 /usr/bin/crontab
2142478  148 -rwsr-xr-x  1 root     shadow      146459 Sep 15  2015 /usr/bin/chsh
2142480  156 -rwsr-xr-x  1 root     shadow      152387 Sep 15  2015 /usr/bin/gpasswd
2142479   48 -rwsr-xr-x  1 root     shadow       46967 Sep 15  2015 /usr/bin/expiry
311484    48 -rwsr-x---  1 root     messagebus   47912 Sep 16  2014 /lib64/dbus-1/dbus-daemon-
launch-helper
876574    36 -rwsr-xr-x  1 root     shadow       35688 Apr 10  2014 /sbin/unix_chkpwd
876648    12 -rwsr-xr-x  1 root     shadow       10736 Dec 16  2011 /sbin/unix2_chkpwd
 49308    68 -rwsr-xr-x  1 root     root         63376 May 27  2015 /opt/likewise/bin/ksu
1130552   40 -rwsr-xr-x  1 root     root         40016 Apr 16  2015 /bin/su
1130511   40 -rwsr-xr-x  1 root     root         40048 Apr 15  2011 /bin/ping
1130600  100 -rwsr-xr-x  1 root     root         94808 Mar 11  2015 /bin/mount
1130601   72 -rwsr-xr-x  1 root     root         69240 Mar 11  2015 /bin/umount
1130512   36 -rwsr-xr-x  1 root     root         35792 Apr 15  2011 /bin/ping6
2012 /lib64/dbus-1/dbus-daemon-launch-helper
```

2   Run the following command to verify that all files on the virtual appliance have an owner.

```
find / -path /proc -prune -o -nouser -o -nogroup
```

3   Review permissions for all files to the virtual appliance to verify that none of them are world writable by running the following command.

```
find / -name "*.*" -type f -perm -a+w | xargs ls -ldb
```

4   Run the following command to verify that only the vcac user owns the correct files.

```
find / -name "proc" -prune -o -user vcac -print | egrep -v -e "*/vcac/*" | egrep
-v -e "*/vmware-vcac/*"
```

If no results appear, then all correct files are owned only by the vcac user.

5   Verify that the following files are writeable only by the vcac user.

```
/etc/vcac/vcac/security.properties
```

```
/etc/vcac/vcac/solution-users.properties
```

```
/etc/vcac/vcac/sso-admin.properties
```

```
/etc/vcac/vcac/vcac.keystore
```

```
/etc/vcac/vcac/vcac.properties
```

Also verify the following files and their sub-directories

```
/var/log/vcac/*
```

```
/var/lib/vcac/*
```

```
/var/cache/vcac/*
```

6 Verify that only the vcac or root user can read the correct files in the following directories and their sub-directories.

```
/etc/vcac/*
```

```
/var/log/vcac/*
```

```
/var/lib/vcac/*
```

```
/var/cache/vcac/*
```

7 Verify that the correct files are owned only by the vco or root user, as shown in in the following directories and their sub-directories.

```
/etc/vco/*
```

```
/var/log/vco/*
```

```
/var/lib/vco/*
```

```
/var/cache/vco/*
```

8 Verify that the correct files are writeable only by the vco or root user, as shown in in the following directories and their sub-directories.

```
/etc/vco/*
```

```
/var/log/vco/*
```

```
/var/lib/vco/*
```

```
/var/cache/vco/*
```

9 Verify that the correct files are readable only by the vco or root user, as shown in in the following directories and their sub-directories.

```
/etc/vco/*
```

```
/var/log/vco/*
```

```
/var/lib/vco/*
```

```
/var/cache/vco/*
```

Customizing Console Proxy Configuration

You can customize the remote console configuration for vRealize Automation to facilitate troubleshooting and organizational practices.

When you install, configure, or maintain vRealize Automation, you can change some settings to enable troubleshooting and debugging of your installation. Catalog and audit each of the changes you make to ensure that applicable components are properly secured according to their required use. Do not proceed to production if you are not sure that your configuration changes are correctly secured.

Customize VMware Remote Console Ticket Expiry

You can customize the validity period for remote console tickets used in establishing VMware Remote Console connections.

When a user makes VMware Remote Console connections, the system creates and returns a one-time credential that establishes a specific connection to a virtual machine. You can set the ticket expiry for a specified time frame in minutes.

Procedure

1   Open the `/etc/vcac/security.properties` file in a text editor.

2   Add a line to the file of the form `consoleproxy.ticket.validitySec=30`.

    In this line the numerical value specifies the number of minutes before the ticket expires.

3   Save the file and close it.

4   Restart the vcac-server using the command `/etc/init.d/vcac-server restart`.

The ticket expiry value is reset to the specified time frame in minutes.

Customize Console Proxy Server Port

You can customize the port on which the VMware Remote Console console proxy listens for messages.

Procedure

1   Open the `/etc/vcac/security.properties` file in a text editor.

2   Add a line to the file of the form `consoleproxy.service.port=8445`.

    The numerical value specifies the console proxy service port number, in this case 8445.

3   Save the file and close it.

4   Restart the vcac-server using the command `/etc/init.d/vcac-server restart`.

The proxy service port changes to the specified port number.

Configure X-XSS-Protection Response Header

Add the X-XSS-Protection response header to the haproxy configuration file.

Procedure

1 Open `/etc/haproxy/conf.d/20-vcac.cfg` for editing.

2 Add the following lines in a front end section:

```
rspdel X-XSS-Protection:\ 1;\ mode=block
            rspadd X-XSS-Protection:\ 1;\ mode=block
```

3 Reload the HAProxy configuration using the following command.

`/etc/init.d/haproxy reload`

Configure HTTP Strict Transport Security Response Header

Add the HTTP Strict Transport (HSTS) response header to the HAProxy configuration.

Procedure

1 Open `/etc/haproxy/conf.d/20-vcac.cfg` for editing.

2 Add the following lines in a front end section:

```
rspdel Strict-Transport-Security:\ max-age=31536000
        rspadd Strict-Transport-Security:\ max-age=31536000
```

3 Reload the HAProxy configuration using the following command.

`/etc/init.d/haproxy reload`

Configure X-Frame-Options Response Header

The X-Frame-Options response header may appear twice in some cases.

The X-Frame-Options response header may appear twice because the vIDM service adds this header to the back end as well as to HAProxy. You can prevent it appearing twice with an appropriate configuration.

Procedure

1 Open `/etc/haproxy/conf.d/20-vcac.cfg` for editing.

2 Locate the following line in the front end section:

`rspadd X-Frame-Options:\ SAMEORIGIN`

3 Add the following lines before the line you located in the preceding step:

`rspdel X-Frame-Options:\ SAMEORIGIN`

4 Reload the HAProxy configuration using the following command.

`/etc/init.d/haproxy reload`

Configuring Server Response Headers

As a security best practice, configure your vRealize Automation system to limit information available to potential attackers.

To the extent possible, minimize the amount of information that your system shares about its identity and version. Hackers and malicious actors can use this information to craft targeted attacks against your Web server or version.

Configure the Lighttpd Server Response Header

As a best practice, create a blank server header for the vRealize Automation appliance lighttpd server.

Procedure

1   Open the `/opt/vmware/etc/lighttpd/lighttpd.conf` file in a text editor.

2   Add the `server.tag = " "` to the file.

3   Save your changes and close the file.

4   Restart the lighttpd server by running the `# /opt/vmware/etc/init.d/vami-lighttp restart` command.

Configure the TCServer Response Header for the vRealize Automation Appliance

As a best practice, create a custom blank server header for the TCServer response header used with the vRealize Automation appliance to limit the possibility of a malicious attacker obtaining valuable information.

Procedure

1   Open the `/etc/vco/app-server/server.xml` file in a text editor.

2   In each `<Connector>` element add `server=" "`.

    For example: `<Connector protocol="HTTP/1.1" server="" ........ />`

3   Save your changes and close the file.

4   Restart the server using the following command.

    `service vco-server restart`

Configure the Internet Information Services Server Response Header

As a best practice, create a custom blank server header for the Internet Information Services (IIS) server used with the Identity Appliance to limit the possibility of malicious attackers obtaining valuable information.

Procedure

1   Open the `C:\Windows\System32\inetsrv\urlscan\UrlScan.ini` file in a text editor.

2   Search for `RemoveServerHeader=0` and change it to `RemoveServerHeader=1.`

3   Save your changes and close the file.

4   Restart the server by running the `iisreset` command.

What to do next

Disable the IIS X-Powered By header by removing HTTP Response headers from the list in the IIS Manager Console.

1   Open the IIS Manager console.

2   Open the HTTP Response Header and remove it from the list.

3   Restart the server by running the `iisreset` command.

Set vRealize Automation appliance Session Timeout

Configure the session timeout setting on the vRealize Automation appliance in accordance with your company security policy.

The vRealize Automation appliance default session timeout on user inactivity is 30 minutes. To adjust this time out value to conform to your organization's security policy, edit the `web.xml` file on your vRealize Automation appliance host machine.

Procedure

1   Open the `/usr/lib/vcac/server/webapps/vcac/WEB-INF/web.xml` file in a text editor.

2   Find `session-config` and set the session-timeout value. See the following code sample.

```
<!-- 30 minutes session expiration time -->
  <session-config>
        <session-timeout>30</session-timeout>
        <tracking-mode>COOKIE</tracking-mode>
        <cookie-config>
              <path>/</path>
        </cookie-config>
  </session-config>
```

3   Restart the server by running the following command.

```
service vcac-server restart
```

Managing Nonessential Software

To minimize security risks, remove or configure nonessential software from your vRealize Automation host machines.

Configure all software that you do not remove in accordance with manufacturer recommendations and security best practices to minimize its potential to create security breaches.

Secure the USB Mass Storage Handler

Secure the USB mass storage handler to prevent its use as the USB device handler with the VMware virtual appliance host machines. Potential attackers can exploit this handler to compromise your system.

Procedure

1 Open the `/etc/modprobe.conf.local` file in a text editor.

2 Ensure that the `install usb-storage /bin/true` line appears in the file.

3 Save the file and close it.

Secure the Bluetooth Protocol Handler

Secure the Bluetooth Protocol Handler on your virtual appliance host machines to prevent potential attackers from exploiting it.

Binding the Bluetooth protocol to the network stack is unnecessary and can increase the attack surface of the host.

Procedure

1 Open the `/etc/modprobe.conf.local` file in a text editor.

2 Ensure that the following line appears in this file.

    `install bluetooth /bin/true`

3 Save the file and close it.

Secure the Stream Control Transmission Protocol

Prevent the Stream Control Transmission Protocol (SCTP) from loading on your system by default. Potential attackers could exploit this protocol to compromise your system.

Configure your system to prevent the Stream Control Transmission Protocol (SCTP) module from loading unless it is absolutely necessary. SCTP is an unused IETF-standardized transport layer protocol. Binding this protocol to the network stack increases the attack surface of the host. Unprivileged local processes could cause the kernel to dynamically load a protocol handler by opening a socket using the protocol.

Procedure

1 Open the `/etc/modprobe.conf.local` file in a text editor.

2 Ensure that the following line appears in this file.

    `install sctp /bin/true`

3 Save the file and close it.

Secure the Datagram Congestion Protocol

As part of your system hardening activities, prevent the Datagram Congestion Protocol (DCCP) from loading on your virtual appliance host machines by default. Potential attackers can exploit this protocol to compromise your system.

Avoid loading the Datagram Congestion Control Protocol (DCCP) module, unless it is absolutely necessary. DCCP is a proposed transport layer protocol, which is not used. Binding this protocol to the network stack increases the attack surface of the host. Unprivileged local processes can cause the kernel to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1 Open the /etc/modprobe.conf.local file in a text editor.

2 Ensure that the DCCP lines appear in the file.

```
install dccp/bin/true
 install dccp_ipv4/bin/true
 install dccp_ipv6/bin/true
```

3 Save the file and close it.

Secure Network Bridging

Prevent the network bridging module from loading on your system by default. Potential attackers could exploit it to compromise your system.

Configure your system to prevent the network from loading, unless it is absolutely necessary. Potential attackers could exploit it to bypass network partitioning and security.

Procedure

1 Run the following command on all VMware virtual appliance host machines.

```
# rmmod bridge
```

2 Open the /etc/modprobe.conf.local file in a text editor.

3 Ensure that the following line appears in this file.

```
install bridge /bin/false
```

4 Save the file and close it.

Secure Reliable Datagram Sockets Protocol

As part of your system hardening activities, prevent the Reliable Datagram Sockets Protocol (RDS) from loading on your virtual appliance host machines by default. Potential attackers can exploit this protocol to compromise your system.

Binding the Reliable Datagram Sockets (RDS) Protocol to the network stack increases the attack surface of the host. Unprivileged local processes can cause the system to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1   Open the `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the `install rds /bin/true` line appears in this file.

3   Save the file and close it.

Secure Transparent Inter-Process Communication Protocol

As part of your system hardening activities, prevent the Transparent Inter-Process Communication Protocol (TIPC) from loading on your virtual appliance host machines by default. Potential attackers can exploit this protocol to compromise your system.

Binding the Transparent Inter-Process Communications (TIPC) Protocol to the network stack increases the attack surface of the host. Unprivileged local processes can cause the kernel to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1   Open the `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the `install tipc /bin/true` line appears in this file.

3   Save the file and close it.

Secure Internetwork Packet Exchange Protocol

Prevent the Internetwork Packet Exchange Protocol (IPX) from loading on your system by default. Potential attackers could exploit this protocol to compromise your system.

Avoid loading the Internetwork Packet Exchange (IPX) Protocol module unless it is absolutely necessary. IPX protocol is an obsolete network-layer protocol. Binding this protocol to the network stack increases the attack surface of the host. Unprivileged local processes could cause the system to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1   Open the `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the following line appears in this file.

```
install ipx /bin/true
```

3   Save the file and close it.

Secure Appletalk Protocol

Prevent the Appletalk Protocol from loading on your system by default. Potential attackers could exploit this protocol to compromise your system.

Avoid loading the Appletalk Protocol module unless it is absolutely necessary. Binding this protocol to the network stack increases the attack surface of the host. Unprivileged local processes could cause the system to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1   Open the `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the following line appears in this file.

    install appletalk /bin/true

3   Save the file and close it.

Secure DECnet Protocol

Prevent the DECnet Protocol from loading on your system by default. Potential attackers could exploit this protocol to compromise your system.

Avoid loading the DECnet Protocol module unless it is absolutely necessary. Binding this protocol to the network stack increases the attack surface of the host. Unprivileged local processes could cause the system to dynamically load a protocol handler by using the protocol to open a socket.

Procedure

1   Open the DECnet Protocol `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the following line appears in this file.

    install decnet /bin/true

3   Save the file and close it.

Secure Firewire Module

Prevent the Firewire module from loading on your system by default. Potential attackers could exploit this protocol to compromise your system.

Avoid loading the Firewire module unless it is absolutely necessary.

Procedure

1   Open the `/etc/modprobe.conf.local` file in a text editor.

2   Ensure that the following line appears in this file.

    install ieee1394 /bin/true

3   Save the file and close it.

Securing the Infrastructure as a Service Component

When you harden your system, secure the vRealize Automation Infrastructure as a Service (IaaS) component and its host machine to prevent potential attackers from exploiting it.

You must configure security setting for the vRealize Automation Infrastructure as a Service (IaaS) component and the host on which it resides. You must set or verify the configuration of other related components and applications. In some cases, you can verify existing settings, in others you must change or add settings for an appropriate configuration.

Disabling Windows Time Service

As a security best practice, use authorized time servers rather than host time synchronization in a vRealize Automation production environment.

In a production environment, disable host time synchronization and use authorized time servers to support accurate tracking of user actions, and identification of potential malicious attacks and intrusion through auditing and logging.

Configuring TLS for Infrastructure as a Service Data-in-Transit

Ensure that your vRealize Automation deployment uses strong TLS protocols to secure transmission channels for Infrastructure as a Service components.

Secure Sockets Layer (SSL) and the more recently developed Transport Layer Security (TLS) are cryptographic protocols that help ensure system security during network communications between different system components. As SSL is an older standard, many of its implements no longer provide adequate security against potential attacks. Serious weaknesses have been identified with earlier SSL protocols, including SSLv2 and SSLv3. These protocols are no longer considered secure.

Depending on your organization's security policies you may wish to also disable TLS 1.0.

Note   When terminating TLS at the load balancer, also disable weak protocols such as SSLv2, SSLv3, as well as TLS 1.0 if required.

Disable SSLv3 in Internet Information Services

As a security best practice, disable SSLv3 in Internet Information Services (IIS) on the Infrastructure as a Service (IaaS) host server machine.

Procedure

1   Run the Windows registry editor as an administrator.

2   Navigate to
    `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\Schannel\Protocols\` in the registry window.

3   Right-click on **Protocols**, and select **New > Key**.

4   Enter `SSL 3.0`.

5   In the navigation tree, right-click the newly created **SSL 3.0** key, and in the pop-up menu select **New > Key** and enter `Client`.

6 In the navigation tree, right-click on the newly created **SSL 3.0** key, and in the pop-up menu select **New > Key** and enter `Server`.

7 In the navigation tree, under SSL 3.0, right-click **Client**, and select **New > DWORD (32-bit) Value** and enter `DisabledByDefault`.

8 In the navigation tree, under SSL 3.0, select **Client**, and in the right pane, double-click **DisabledByDefault** and enter `1`.

9 In the navigation tree, under SSL 3.0, right-click **Server**, and select **New > DWORD (32-bit) Value** and enter `Enabled`.

10 In the navigation tree, under SSL 3.0, select **Server**, and in the right pane, double-click the enabled **DWORD** and enter `0`.

11 Restart the Windows Server.

Disable TLS 1.0 for IaaS

To provide maximum security, configure IaaS to use pooling and disable TLS 1.0.

For more information, see the Microsoft knowledge base article https://support.microsoft.com/en-us/kb/245030.

Procedure

1 Configure IaaS to use pooling instead of web sockets.

   a Update the Manager Services configuration file `C:\Program Files (x86)\VMware\vCAC\Server\ManagerService.exe.config` by adding the following values in the <appSettings> section

```
<add key="Extensibility.Client.RetrievalMethod" value="Polling"/>
<add key="Extensibility.Client.PollingInterval" value="2000"/>
<add key="Extensibility.Client.PollingMaxEvents" value="128"/>
```

   b Restart the Manager Service (VMware vCloud Automation Center Service).

2 Verify that TLS 1.0 is disabled on the IaaS server.

   a Run the registry editor as an administrator.

   b In the registry window, navigate to `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\Schannel\Protocols\`

   c Right-click on Protocols and select **New > Key** and then enter `TLS 1.0`.

   d In the navigation tree, right-click on the TLS 1.0 key that you just created, and in the pop-up menu select **New > Key** and enter `Client`.

   e In the navigation tree, right-click on the TLS 1.0 key that you just created and in the pop up menu select **New > Key** and enter `Server`.

f   In the navigation tree, under TLS 1.0, right-click on **Client**, and then click **New > DWORD (32-bit) Value**and enter `DisabledByDefault`.

g   In the navigation tree, under TLS 1.0, select **Client**, and in the right pane, double-click **DisabledByDefault** DWORD and enter `1`.

h   In the navigation tree, under TLS 1.0, right-click **Server**, and select **New > DWORD (32-bit) Value** and enter `Enabled`.

i   In the navigation tree, under TLS 1.0, select **Server**, and in the right pane, double-click **Enabled** DWORD and enter `0`.

j   Restart the Windows Server.

Configuring TLS Cipher Suites

For maximum security, you must configure vRealize Automation components to use strong ciphers. The encryption cipher negotiated between the server and the browser determines the encryption strength that is used in a TLS session. To ensure that only strong ciphers are selected, disable weak ciphers in vRealize Automation components. Configure the server to support only strong ciphers and to use sufficiently large key sizes. Also, configure all ciphers in a suitable order.

Cipher Suites that are not Acceptable

Disable cipher suites that do not offer authentication such as NULL cipher suites, aNULL, or eNULL. Also disable anonymous Diffie-Hellman key exchange (ADH), export level ciphers (EXP, ciphers containing DES), key sizes smaller than 128 bits for encrypting payload traffic, the use of MD5 as a hashing mechanism for payload traffic, IDEA Cipher Suites, and RC4 cipher suites. Also ensure that cipher suites using Diffie-Hellman (DHE) key exchange are disabled.

Verifying Host Server Security

As a security best practice, verify the security configuration of your Infrastructure as a Service (IaaS) host server machines.

Microsoft supplies several tools to help you verify security on host server machines. Contact your Microsoft vendor for guidance on the most appropriate use of these tools.

Verify Host Server Secure Baseline

Run the Microsoft Baseline Security Analyzer (MBSA) to quickly confirm that your server has the latest updates or hot fixes. You can use the MBSA to install missing security patches from Microsoft to keep your server up-to-date with Microsoft security recommendations.

Download the latest version of the MBSA tool from the Microsoft website.

Verify Host Server Security Configuration

Use the Windows Security Configuration Wizard (SCW) and the Microsoft Security Compliance Manager (SCM) toolkit to verify that the host server is securely configured.

Run the SCW from the administrative tools from your Windows server. This tool can identify the roles of your server and the installed features including networking, Windows firewalls, and registry settings. Compare the report with the latest hardening guidance from the relevant SCM for your Windows server. Based on the results, you can fine tune security settings for each feature such as network services, account settings, and Windows firewalls, and apply the settings to your server.

You can find more information about the SCW tool on the Microsoft Technet Web site.

Protecting Application Resources

As a security best practice, ensure that all relevant Infrastructure as a Service files have the appropriate permissions.

Review Infrastructure as a Service files against your Infrastructure as a Service installation. In most cases, subfolders and files for every folder should have the same settings as the folder.

Directory or File	Group or Users	Full Control	Modify	Read & Execute	Read	Write
VMware\vCAC\Agents \<agent_name> \logs	SYSTEM	X	X	X	X	X
	Administrator	X	X	X	X	X
	Administrators	X	X	X	X	X
VMware\vCAC\Agents\ <agent_name> \temp	SYSTEM	X	X	X	X	X
	Administrator	X	X	X	X	X
	Administrators	X	X	X	X	X
VMware\vCAC\Agents\	SYSTEM	X	X	X	X	X
	Administrators	X	X	X	X	X
	Users			X	X	
VMware\vCAC\Distributed Execution Manager\	SYSTEM	X	X	X	X	X
	Administrators	X	X	X	X	X
	Users			X	X	
VMware\vCAC\Distributed Execution Manager\DEM\Logs	SYSTEM	X	X	X	X	X
	Administrator	X	X	X	X	X
	Administrators	X	X	X	X	X
VMware\vCAC\Distributed Execution Manager\DEO\Logs	SYSTEM	X	X	X	X	X
	Administrator	X	X	X	X	X
	Administrators	X	X	X	X	X
VMware\vCAC\Management Agent\	SYSTEM	X	X	X	X	X
	Administrators	X	X	X	X	X
	Users			X	X	
VMware\vCAC\Server\	SYSTEM	X	X	X	X	X
	Administrators	X	X	X	X	X

Directory or File	Group or Users	Full Control	Modify	Read & Execute	Read	Write
	Users			X	X	
VMware\vCAC\Web API	SYSTEM	X	X	X	X	X
	Administrators	X	X	X	X	X
	Users			X	X	

Secure the Infrastructure as a Service Host Machine

As a security best practice, review basic settings on your Infrastructure as a Service (IaaS) host machine to ensure that it conforms to security guidelines.

Secure miscellaneous accounts, applications, ports, and services on the Infrastructure as a Service (IaaS) host machine.

Verify Server User Account Settings

Verify that no unnecessary local and domain user accounts and settings exist. Restrict any user account that is not related to the application functions to those required for administration, maintenance, and troubleshooting. Restrict remote access from domain user accounts to the minimum required to maintain the server. Strictly control and audit these accounts.

Delete Unnecessary Applications

Delete all unnecessary applications from the host servers. Unnecessary applications increase the risk of exposure because of their unknown or unpatched vulnerabilities.

Disable Unnecessary Ports and Services

Review the host server's firewall for the list of open ports. Block all ports that are not required for the IaaS component or critical system operation. See Configuring Ports and Protocols. Audit the services running against your host server, and disable those that are not required.

Configuring Host Network Security

<div style="text-align: right; font-size: 2em;">9</div>

To provide maximum protection against known security threats, configure network interface and communication settings on all VMware host machines.

As part of a comprehensive security plan, configure network interface security settings for the VMware virtual appliances and the Infrastructure as a Service components in accordance with established security guidelines.

This chapter includes the following topics:

- Configuring Network Settings for VMware Appliances
- Configuring Network Settings for the Infrastructure as a Service Host
- Configuring Ports and Protocols

Configuring Network Settings for VMware Appliances

To ensure that your VMware virtual appliance host machines support only safe and essential communications, review and edit their network communication settings.

Examine the network IP protocol configuration of your VMware host machines, and configure network settings in accordance with security guidelines. Disable all nonessential communication protocols.

Prevent User Control of Network Interfaces

As a security best practice, allow users only the system privileges that they need to do their jobs on VMware appliance host machines.

Permitting user accounts with privileges to manipulate network interfaces can result in bypassing network security mechanisms or denial of service. Restrict the ability to change network interface settings to privileged users.

Procedure

1 Run the following command on each VMware appliance host machine.

```
# grep -i '^USERCONTROL=' /etc/sysconfig/network/ifcfg*
```

2 Make sure that each interface is set to NO.

Set TCP Backlog Queue Size

To provide some level of defense against malicious attacks, configure a default TCP backlog queue size on VMware appliance host machines.

Set the TCP backlog queue sizes to an appropriate default size to provide mitigation for TCP denial or service attacks. The recommended default setting is 1280.

Procedure

1   Run the following command on each VMware appliance host machine.

    # cat /proc/sys/net/ipv4/tcp_max_syn_backlog

2   Open the /etc/sysctl.conf file in a text editor.

3   Set the default TCP backlog queue size by adding the following entry to the file.

    net.ipv4.tcp_max_syn_backlog=1280

4   Save your changes and close the file.

Deny ICMPv4 Echoes to Broadcast Address

As a security best practice, verify that your VMware appliance host machines ignore ICMP broadcast address echo requests.

Responses to broadcast Internet Control Message Protocol (ICMP) echoes provide an attack vector for amplification attacks and can facilitate network mapping by malicious agents. Configuring your appliance host machines to ignore ICMPv4 echoes provides protection against such attacks.

Procedure

1   Run the # cat /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts command on the VMware virtual appliance host machines to confirm that they deny IPv4 broadcast address echo requests.

    If the host machines are configured to deny IPv4 redirects, this command will return a value of 0 for /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts.

2   To configure a virtual appliance host machine to deny ICMPv4 broadcast address echo requests, open the /etc/sysctl.conf file on Windows host machines in a text editor.

3   Locate the entry that reads net.ipv4.icmp_echo_ignore_broadcasts=0 . If the value for this entry is not set to zero or if the entry does not exist, add it or update the existing entry accordingly.

4   Save the changes and close the file.

Disable IPv4 Proxy ARP

Verify that IPv4 Proxy ARP is disabled if not otherwise required on your VMware appliance host machines to prevent unauthorized information sharing.

IPv4 Proxy ARP allows a system to send responses to ARP requests on one interface on behalf of hosts connected to another interface. Disable it if not needed to prevent leakage of addressing information between the attached network segments.

Procedure

1   Run the # `grep [01] /proc/sys/net/ipv4/conf/*/proxy_arp|egrep "default|all"` command on the VMware virtual appliance host machines to verify that IPv4 Proxy ARP is disabled.

    If IPv6 Proxy ARP is disabled on the host machines, this command will return values of 0.

    ```
    /proc/sys/net/ipv4/conf/all/proxy_arp:0
    /proc/sys/net/ipv4/conf/default/proxy_arp:0
    ```

    If the host machines are configured correctly, no further action is necessary.

2   If you need to configure IPv6 Proxy ARP on host machines, open the `/etc/sysctl.conf` file in a text editor.

3   Check for the following entries.

    ```
    net.ipv4.conf.default.proxy_arp=0
    net.ipv4.conf.all.proxy_arp=0
    ```

    If the entries do not exist or if their values are not set to zero, add the entries or update the existing entries accordingly.

4   Save any changes you made and close the file.

Deny IPv4 ICMP Redirect Messages

As a security best practice, verify that your VMware virtual appliance host machines deny IPv4 ICMP redirect messages.

Routers use ICMP redirect messages to tell hosts that a more direct route exists for a destination. A malicious ICMP redirect message can facilitate a man-in-the-middle attack. These messages modify the host's route table and are unauthenticated. Ensure that your system is configured to ignore them if they are not otherwise needed.

Procedure

1   Run the # `grep [01] /proc/sys/net/ipv4/conf/*/accept_redirects|egrep "default|all"` command on the VMware appliance host machines to confirm that they deny IPv4 redirect messages.

    If the host machines are configured to deny IPv4 redirects, this command returns the following:

    /proc/sys/net/ipv4/conf/all/accept_reidrects:0

    /proc/sys/net/ipv4/conf/default/accept_redirects:0

2   If you need to configure a virtual appliance host machine to deny IPv4 redirect messages, open the `/etc/sysctl.conf` file in a text editor.

3   Check the values of the lines that begin with `net.ipv4.conf`.

If the values for the following entries are not set to zero or if the entries do not exist, add them to the file or update the existing entries accordingly.

```
net.ipv4.conf.all.accept_redirects=0
net.ipv4.conf.default.accept_redirects=0
```

4   Save the changes you made and close the file.

Deny IPv6 ICMP Redirect Messages

As a security best practice, verify that your VMware virtual appliance host machines deny IPv6 ICMP redirect messages.

Routers use ICMP redirect messages to tell hosts that a more direct route exists for a destination. A malicious ICMP redirect message can facilitate a man-in-the-middle attack. These messages modify the host's route table and are unauthenticated. Ensure your system is configured to ignore them if they not otherwise needed.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv6/conf/*/accept_redirects|egrep "default|all"` command on the VMwarevirtual appliance host machines to confirm that they deny IPv6 redirect messages.

If the host machines are configured to deny IPv6 redirects, this command returns the following:

`/proc/sys/net/ipv6/conf/all/accept_redirects:0`

`/proc/sys/net/ipv6/conf/default/accept_redirects:0`

2   To configure a virtual appliance host machine to deny IPv4 redirect messages, open the `/etc/sysctl.conf` file in a text editor.

3   Check the values of the lines that begin with `net.ipv6.conf`.

If the values for the following entries in the are not set to zero or if the entries do not exist, add them to the file or update the existing entries accordingly.

```
net.ipv6.conf.all.accept_redirects=0
net.ipv6.conf.default.accept_redirects=0
```

4   Save the changes and close the file.

Log IPv4 Martian Packets

As a security best practice, verify that your VMware virtual appliance host machines log IPv4 Martian packets.

Martian packets contain addresses that the system knows to be invalid. Configure your host machines to log these messages so that you can identify misconfigurations or attacks in progress.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv4/conf/*/log_martians|egrep "default|all"` command on the VMware appliance host machines to verify that they log IPv4 Martian packets.

    If the virtual machines are configured to log Martian packers, they return the following:

    ```
    /proc/sys/net/ipv4/conf/all/log_martians:1
    /proc/sys/net/ipv4/default/log_martians:1
    ```

    If the host machines are configured correctly, no further action is necessary.

2   If you need to configure virtual machines to log IPv4 martian packets, open the `/etc/sysctl.conf` file in a text editor.

3   Check the values of the lines that start with `net.ipv4.conf`.

    If the value for the following entries are not set to 1or if they do not exist, add them to the file or update the existing entries accordingly.

    ```
    net.ipv4.conf.all.log_martians=1
    net.ipv4.conf.default.log_martians=1
    ```

4   Save your changes and close the file.

Use IPv4 Reverse Path Filtering

As a security best practice, verify that your VMware virtual appliance host machines use IPv4 reverse path filtering.

Reverse-path filtering protects against spoofed source addresses by causing the system to discard packets with source addresses that have no route or a route that does not point towards the originating interface. Configure your host machines to use reverse-path filtering whenever possible. In some cases, depending on the system role, reverse-path filtering can cause the system to discard legitimate traffic. If you encounter such problems, you might need to use a more permissive mode or disable reverse-path filtering altogether.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv4/conf/*/rp_filter|egrep "default|all"` command on the VMware virtual appliance host machines to verify that they use IPv4 reverse path filtering.

    If the virtual machines use IPv4 reverse path filtering, this command returns the following:

    ```
    /proc/sys/net/ipv4/conf/all/rp_filter:1
    /proc/sys/net/ipv4/conf/default/re_filter:1
    ```

    If your virtual machines are configured correctly, no further action is required.

2   If you need to configure IPv4 reverse path filtering on host machines, open the `/etc/sysctl.conf` file in a text editor.

3 Check the values of the lines that begin with `net.ipv4.conf`.

If the values for the following entries are not set to 1 or if they do not exist, add them to the file or update the existing entries accordingly.

```
net.ipv4.conf.all.rp_filter=1
net.ipv4.conf.default.rp_filter=1
```

4 Save the changes and close the file.

Deny IPv4 Forwarding

Verify that your VMware appliance host machines deny IPv4 forwarding.

If the system is configured for IP forwarding and is not a designated router, attackers could use it to bypass network security by providing a path for communication not filtered by network devices. Configure your virtual appliance host machines to deny IPv4 forwarding to avoid this risk.

Procedure

1 Run the `# cat /proc/sys/net/ipv4/ip_forward` command on the VMware appliance host machines to confirm that they deny IPv4 forwarding.

If the host machines are configured to deny IPv4 forwarding, this command will return a value of 0 for `/proc/sys/net/ipv4/ip_forward`. If the virtual machines are configured correctly, no further action is necessary.

2 To configure a virtual appliance host machine to deny IPv4 forwarding, open the `/etc/sysctl.conf` file in a text editor.

3 Locate the entry that reads `net.ipv4.ip_forward=0`. If the value for this entry is not currently set to zero or if the entry does not exist, add it or update the existing entry accordingly.

4 Save any changes and close the file.

Deny IPv6 Forwarding

As a security best practice, verify that your VMware appliance host systems deny IPv6 forwarding.

If the system is configured for IP forwarding and is not a designated router, attackers could use it to bypass network security by providing a path for communication not filtered by network devices. Configure your virtual appliance host machines to deny IPv6 forwarding to avoid this risk.

Procedure

1 Run the `# grep [01] /proc/sys/net/ipv6/conf/*/forwarding|egrep "default|all"` command on the VMware appliance host machines to verify that they deny IPv6 forwarding.

If the host machines are configured to deny IPv6 forwarding, this command will return the following:

```
/proc/sys/net/ipv6/conf/all/forwarding:0
/proc/sys/net/ipv6/conf/default/forwarding:0
```

If the host machines are configured correctly, no further action is necessary.

2   If you need to configure a host machine to deny IPv6 forwarding, open the `/etc/sysctl.conf` file in a text editor.

3   Check the values of the lines that begin with `net.ipv6.conf`.

If the values for the following entries are not set to zero or if the entries do not exist, add the entries or update the existing entries accordingly.

```
            net.ipv6.conf.all.accept_redirects=0
  net.ipv6.conf.default.accept_redirects=0
```

4   Save any changes you made and close the file.

Use IPv4 TCP Syncookies

Verify that your VMware appliance host machines use IPv4 TCP Syncookies.

A TCP SYN flood attack might cause a denial of service by filling a system's TCP connection table with connections in the SYN_RCVD state. Syncookies prevent tracking a connection until receipt of a subsequent ACK, verifying that the initiator is attempting a valid connection and is not a flood source. This technique does not operate in a fully standards-compliant manner, but is only activated during a flood condition, and allows defence of the system while continuing to service valid requests.

Procedure

1   Run the `# cat /proc/sys/net/ipv4/tcp_syncookies` command on the VMware appliance host machines to verify that they use IPv4 TCP Syncookies.

If the host machines are configured to deny IPv4 forwarding, this command will return a value of 1 for `/proc/sys/net/ipv4/tcp_syncookies`. If the virtual machines are configured correctly, no further action is necessary.

2   If you need to configure a virtual appliance to use IPv4 TCP Syncookies, open the `/etc/sysctl.conf` in a text editor.

3   Locate the entry that reads `net.ipv4.tcp_syncookies=1`.

If the value for this entry is not currently set to one or if it does not exist, add the entry or update the existing entry accordingly.

4   Save any changes you made and close the file.

Deny IPv6 Router Advertisements

Verify that VMware host machines deny the acceptance of router advertisements and ICMP redirects unless otherwise required for system operation.

IPv6 enables systems to configure their networking devices by automatically using information from the network. From a security perspective, manually configuring important configuration information is preferable to accepting it from the network in an unauthenticated way.

Procedure

1 Run the # `grep [01] /proc/sys/net/ipv6/conf/*/accept_ra|egrep "default|all"` command on the VMware appliance host machines to verify that they deny router advertisements.

   If the host machines are configured to deny IPv6 router advertisements, this command will return values of 0:

   ```
   /proc/sys/net/ipv6/conf/all/accept_ra:0
   /proc/sys/net/ipv6/conf/default/accept_ra:0
   ```

   If the host machines are configured correctly, no further action is necessary.

2 If you need to configure a host machine to deny IPv6 router advertisements, open the `/etc/sysctl.conf` file in a text editor.

3 Check for the following entries.

   ```
   net.ipv6.conf.all.accept_ra=0
   net.ipv6.conf.default.accept_ra=0
   ```

   If these entries do not exist, or if their values are not set to zero, add the entries or update the existing entries accordingly.

4 Save any changes you made and close the file.

Deny IPv6 Router Solicitations

As a security best practice, verify that your VMware appliance host machines deny IPv6 router solicitations unless otherwise required for system operation.

The router solicitations setting determines how many router solicitations are sent when bringing up the interface. If addresses are statically assigned, there is no need to send any solicitations.

Procedure

1 Run the # `grep [01] /proc/sys/net/ipv6/conf/*/router_solicitations|egrep "default|all"` command on the VMware appliance host machines to verify that they deny IPv6 router solicitations.

   If the host machines are configured to deny IPv6 router advertisements, this command will return the following:

   ```
   /proc/sys/net/ipv6/conf/all/router_solicitations:0
   /proc/sys/net/ipv6/conf/default/router_solicitations:0
   ```

   If the host machines are configured correctly, no further action is necessary.

2 If you need to configure host machines to deny IPv6 router solicitations, open the `/etc/sysctl.conf` file in a text editor.

3   Check for the following entries.

```
net.ipv6.conf.all.router_solicitations=0
net.ipv6.conf.default.router_solicitations=0
```

If the entries do not exist or if their values are not set to zero, add the entries or update the existing entries accordingly.

4   Save any changes and close the file.

Deny IPv6 Router Preference in Router Solicitations

Verify that your VMware appliance host machines to deny IPv6 router solicitations unless otherwise needed for system operation.

The router preference in the solicitations setting determines router preferences. If addresses are statically assigned, there is no need to receive any router preference for solicitations.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv6/conf/*/accept_ra_rtr_pref|egrep "default|all"` command on the VMware appliance host machines to verify that they deny IPv6 router solicitations.

    If the host machines are configured to deny IPv6 router advertisements, this command will return the following:

```
/proc/sys/net/ipv6/conf/all/accept_ra_rtr_pref:0
/proc/sys/net/ipv6/conf/default/accept_ra_rtr_pref:0
```

    If the host machines are configured correctly, no further action is necessary.

2   If you need to configure host machines to deny IPv6 route solicitations, open the `/etc/sysctl.conf` file in a text editor.

3   Check for the following entries.

```
net.ipv6.conf.all.accept_ra_rtr_pref=0
net.ipv6.conf.default.accept_ra_rtr_pref=0
```

    If the entries do not exist or if their values not set to zero, add the entries or update the existing entries accordingly.

4   Save any changes you made and close the file.

Deny IPv6 Router Prefix

Verify that your VMware appliance host machines deny IPv6 router prefix information unless otherwise required for system operation.

The `accept_ra_pinfo` setting controls whether the system accepts prefix info from the router. If addresses are statically assigned, there is no need to receive any router prefix information.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv6/conf/*/accept_ra_pinfo|egrep "default|all"`
    command on the VMware appliance host machines to verify that they deny IPv6 router prefix
    information.

    If the host machines are configured to deny IPv6 router advertisements, this command will return the
    following.

    ```
    /proc/sys/net/ipv6/conf/all/accept_ra_pinfo:0
    /proc/sys/net/ipv6/conf/default/accept_ra_pinfo:0
    ```

    If the host machines are configured correctly, no further action is necessary.

2   If you need to configure host machines to deny IPv6 router prefix information, open
    the `/etc/sysctl.conf` file in a text editor.

3   Check for the following entries.

    ```
    net.ipv6.conf.all.accept_ra_pinfo=0
    net.ipv6.conf.default.accept_ra_pinfo=0
    ```

    If the entries do not exist or if their values are not set to zero, add the entries or update the existing
    entries accordingly.

4   Save any changes and close the file.

Deny IPv6 Router Advertisement Hop Limit Settings

Verify that your VMware appliance host machines deny IPv6 router hop limit settings unless necessary.

The `accept_ra_defrtr` setting controls whether the system will accept Hop Limit settings from a router
advertisement. Setting it to zero prevents a router from changing your default IPv6 Hop Limit for outgoing
packets.

Procedure

1   Run the `# grep [01] /proc/sys/net/ipv6/conf/*/accept_ra_defrtr|egrep "default|all"`
    command on the VMware appliance host machines to verify that they deny IPv6 router hop limit
    settings.

    If the host machines are configured to deny IPv6 router hop limit settings, this command will return
    values of 0.

    ```
    /proc/sys/net/ipv6/conf/all/accept_ra_defrtr:0
    /proc/sys/net/ipv6/conf/default/accept_ra_defrtr:0
    ```

    If the host machines are configured correctly, no further action is necessary.

2   If you need to configure a host machine to deny IPv6 router hop limit settings, open
    the `/etc/sysctl.conf` file in a text editor.

3 Check for the following entries.

```
net.ipv6.conf.all.autoconf=0
net.ipv6.conf.default.autoconf=0
```

If the entries do not exist or if their values are not set to zero, add the entries or update the existing entries accordingly.

4 Save any changes you made and close the file.

Deny IPv6 Router Advertisement Autoconf Settings

Verify that your VMware appliance host machines deny IPv6 router autoconf settings unless necessary.

The `autoconf` setting controls whether router advertisements can cause the system to assign a global unicast address to an interface.

Procedure

1 Run the `# grep [01] /proc/sys/net/ipv6/conf/*/autoconf|egrep "default|all"` command on the VMware appliance host machines to verify that they deny IPv6 router autoconf settings.

   If the host machines are configured to deny IPv6 router autoconf settings, this command will return values of 0.

```
/proc/sys/net/ipv6/conf/all/autoconf:0
/proc/sys/net/ipv6/conf/default/autoconf:0
```

   If the host machines are configured correctly, no further action is necessary.

2 If you need to configure a host machine to deny IPv6 router autoconf settings, open the `/etc/sysctl.conf` file in a text editor.

3 Check for the following entries.

```
net.ipv6.conf.all.autoconf=0
net.ipv6.conf.default.autoconf=0
```

   If the entries do not exist or if their values are not set to zero, add the entries or update the existing entries accordingly.

4 Save any changes you made and close the file.

Deny IPv6 Neighbor Solicitations

Verify that your VMware appliance host machines to deny IPv6 neighbor solicitations unless necessary.

The `dad_transmits` setting determines how many neighbor solicitations to send out per address (global and link-local) when bringing up an interface to ensure the desired address is unique on the network.

Procedure

1 Run the `# grep [01] /proc/sys/net/ipv6/conf/*/dad_transmits|egrep "default|all"` command on the VMware appliance host machines to confirm that they deny IPv6 neighbor solicitations.

If the host machines are configured to deny IPv6 neighbor solicitations, this command will return values of 0.

```
/proc/sys/net/ipv6/conf/all/dad_transmits:0
/proc/sys/net/ipv6/conf/default/dad_transmits:0
```

If the host machines are configured correctly, no further action is necessary.

2 If you need to configure a host machine to deny IPv6 neighbor solicitations, open the `/etc/sysctl.conf` file in a text editor.

3 Check for the following entries.

```
net.ipv6.conf.all.dad_transmits=0
net.ipv6.conf.default.dad_transmits=0
```

If the entries do not exist or if their values are not set to zero, add the entries or update the existing entries accordingly.

4 Save any changes you made and close the file.

Restrict IPv6 Max Addresses

Verify that your VMware appliance host machines to restrict IPv6 max address settings to the minimum required for system operation.

The max addresses setting determines how many global unicast IPv6 addresses are available to each interface. The default is 16, but you should set to exactly the number of statically configured global addresses required for your system.

Procedure

1 Run the `# grep [1] /proc/sys/net/ipv6/conf/*/max_addresses|egrep "default|all"` command on the VMware appliance host machines to verify that they restrict IPv6 max addresses appropriately.

If the host machines are configured to restrict IPv6 max addresses, this command will return values of 1.

```
/proc/sys/net/ipv6/conf/all/max_addresses:1
/proc/sys/net/ipv6/conf/default/max_addresses:1
```

If the host machines are configured correctly, no further action is necessary.

2 If you need to configure IPv6 max addresses on host machines, open the `/etc/sysctl.conf` file in a text editor.

3   Check for the following entries.

```
net.ipv6.conf.all.max_addresses=1
net.ipv6.conf.default.max_addresses=1
```

If the entries do not exist or if their values are not set to 1, add the entries or update the existing entries accordingly.

4   Save any changes you made and close the file.

Configuring Network Settings for the Infrastructure as a Service Host

As a security best practice, configure network communication settings on your VMware Infrastructure as a Service (IaaS) component host machine according to VMware requirements and guidelines.

Configure the Infrastructure as a Service (IaaS) host machine's network configuration to support full vRealize Automation functions with appropriate security.

See Securing the Infrastructure as a Service Component.

Configuring Ports and Protocols

As a security best practice, configure ports and protocols for all vRealize Automation appliances and components in accordance with VMware guidelines.

Configure incoming and outgoing ports for vRealize Automation components as required for critical system components to operate in production. Disable all unneeded ports and protocols. See *vRealize Automation Reference Architecture*.

User Required Ports

As a security best practice, configure vRealize Automation user ports according to VMware guidelines.

Expose required ports only over a secure network.

SERVER	PORTS
vRealize Automation Appliance	443, 8443

Administrator Required Ports

As a security best practice, configure vRealize Automation administrator ports according to VMware guidelines.

Expose required ports only over a secure network.

SERVER	PORTS
vRealize Application Services Server	5480

vRealize Automation Appliance Ports

As a security best practice, configure incoming and outgoing ports for the vRealize Automation appliance according to VMware recommendations.

Incoming Ports

Configure the minimum required incoming ports for the vRealize Automation appliance. Configure optional ports if needed for your system configuration.

Table 9-1. Minimum Required Incoming Ports

PORT	PROTOCOL	COMMENTS
443	TCP	Access to the vRealize Automation console and API calls.
8443	TCP	Console Proxy (VMRC).
5480	TCP	Access to the virtual appliance Web Management Console.
5488, 5489	TCP	Internal. Used by the vRealize Automation appliance for updates.
5672	TCP	RabbitMQ messaging. **Note** When you cluster vRealize Automation appliance instances, you might need to configure the open ports 4369 and 25672.
40002	TCP	Required for vIDM service. This is firewalled to all external traffic with the exception of traffic from other vRealize Automation appliance nodes when added in HA configuration.

If necessary, configure optional incoming ports.

Table 9-2. Optional Incoming Ports

PORT	PROTOCOL	COMMENTS
22	TCP	(Optional) SSH. In a production environment, disable the SSH service listening on port 22, and close port 22 .
80	TCP	(Optional) Redirects to 443.

Outgoing Ports

Configure the required outgoing ports.

Table 9-3. Minimum Required Outgoing Ports

PORT	PROTOCOL	COMMENTS
25,587	TCP, UDP	SMTP for sending outbound notification emails.
53	TCP, UDP	DNS.
67, 68, 546, 547	TCP, UDP	DHCP.
110, 995	TCP, UDP	POP for receiving inbound notification emails.
143, 993	TCP, UDP	IMAP for receiving inbound notification emails.
443	TCP	Infrastructure as a Service Manager Service over HTTPS.

If necessary, configure optional outgoing ports.

Table 9-4. Optional Outgoing Ports

PORT	PROTOCOL	COMMENTS
80	TCP	(Optional) For fetching software updates. You can download and apply updates separately.
123	TCP, UDP	(Optional) For connecting directly to NTP instead of using host time.

Infrastructure as a Service Ports

As a security best practice, configure incoming and outgoing ports for the Infrastructure as a Service (IaaS) components according to VMware guidelines.

Incoming Ports

Configure the minimum required incoming ports for the IaaS components.

Table 9-5. Minimum Required Incoming Ports

COMPONENT	PORT	PROTOCOL	COMMENTS
Manager Service	443	TCP	Communication with IaaS components and vRealize Automation Appliance over HTTPS. Any virtualization hosts that proxy agents manage must also have TCP port 443 open for incoming traffic

Outgoing Ports

Configure the minimum required outgoing ports for the IaaS components.

Table 9-6. Minimum Required Outgoing Ports

COMPONENT	PORT	PROTOCOL	COMMENTS
All	53	TCP, UDP	DNS.
All		TCP, UDP	DHCP.
Manager Service	443	TCP	Communication with vRealize Automation Appliance over HTTPS.
Web site	443	TCP	Communication with Manager Service over HTTPS.
Distributed Execution Managers	443	TCP	Communication with Manager Service over HTTPS.
Proxy Agents	443	TCP	Communication with Manager Service and virtualization hosts over HTTPS.
Guest Agent	443	TCP	Communication with Manager Service over HTTPS.
Manager Service, Web site	1433	TCP	MSSQL.

If needed, configure optional outgoing ports.

Table 9-7. Optional Outgoing Ports

COMPONENT	PORT	PROTOCOL	COMMENTS
All	123	TCP, UDP	NTP is optional.

Auditing and Logging

As a security best practice, set up auditing and logging on your vRealize Automation system in accordance with VMware recommendations.

Remote logging to a central log host provides a secure store for log files. By gathering log files to a central host, you can monitor the environment with a single tool. Also, you can perform aggregate analysis and search for evidence of threats such as coordinated attacks on multiple entities within the infrastructure. Logging to a secure, centralized log server can help prevent log tampering, and also provides a long-term audit record.

Ensure That the Remote Logging Server Is Secure

Often, after attackers breach the security of your host machine, they attempt to search for and tamper with log files to cover their tracks and maintain control without being discovered. Securing the remote logging server appropriately helps to discourage log tampering.

Use an Authorized NTP Server

Ensure that all host machines use the same relative time source, including the relevant localization offset, and that you can correlate the relative time source to an agreed-upon time standard such as Coordinated Universal Time (UTC). A disciplined approach to time sources enables you to quickly track and correlate an intruder's actions when you review the relevant log files. Incorrect time settings can make it difficult to inspect and correlate log files to detect attacks and can make auditing inaccurate.

Use at least three NTP servers from outside time sources or configure a few local NTP servers on a trusted network that in turn obtain their time from at least three outside time sources.