Anti-Virus for VMware Tanzu v1.4

Anti-Virus for VMware Tanzu 1.4

You can find the most up-to-date technical documentation on the VMware website at:
https://docs.vmware.com/

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

ClamAV Add-on for PCF

This topic is an overview of ClamAV Add-on for Pivotal Cloud Foundry (PCF).

About ClamAV Add-on for PCF

The ClamAV Add-on might be necessary for regulatory purposes if your compliance auditor requires antivirus protection within your PCF environment.

For example, auditors sometimes expect that antivirus protection is present in an environment that must comply with standards such as Payment Card Industry Data Security Standard (PCI DSS) or Health Insurance Portability and Accountability Act (HIPAA).

ClamAV Add-on for PCF complies with the U.S. Department of Defense STIG rule SV-92701r1_rule, version UBTU-16-030900, which belongs to group SRG-OS-000480-GPOS-00227.

Product Snapshot

The following table provides version and version-support information about the ClamAV Add-on for PCF.

Element	Details
Version	1.4.45
Release date	August 14, 2019
Compatible Ops Manager versions	2.3, 2.4, 2.5, 2.6, and 2.7
Compatible Pivotal Application Service (PAS) versions	2.3, 2.4, 2.5, 2.6, and 2.7
Compatible Enterprise Pivotal Container Service (Enterprise PKS) versions	1.2 and later
Compatible BOSH stemcells	Ubuntu (Xenial and Trusty) and Windows (2019, 1803, and 2016)
IaaS support	vSphere, GCP, AWS, Azure, and OpenStack

Features

- Ability to scan VMs and containers for foundations with PAS and Enterprise PKS.
- Supports scheduled scans to reduce workload during peak operation hours.
- Permits adding known signatures to an allowlist.
- Allows you to configure CPU and memory usage limits on VMs of the foundation.

Feedback

If you have a feature request, questions, or information about an issue, please email Pivotal Cloud Foundry Feedback

Release Notes

This topic contains release notes for ClamAV Add-on for PCF.

For product versions and upgrade paths, see Upgrade Planner.

v1.4.46

Release Date: August 30, 2019

New features and changes in this release:

- The bundled ClamAV open source distribution is now v0.101.4.

Security Fixes

This release includes the following security fixes:

- Critical CVE-2019-12900: BZ2_decompress in decompress.c in bzip2 through 1.0.6 has an out-of-bounds write when there are many selectors.
- ClamAV v0.101.4 introduces a scan time limit. This limit now resolves ClamAV's "zip bomb" vulnerability.

For the ClamAV v0.101.4 release notes, see the ClamAV blog.

Known Issue:

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.45

Release Date: August 14, 2019

Features

New features and changes in this release:

- The bundled ClamAV open source distribution is now v0.101.3.

Security Fixes

This release includes the following security fix:

- High CVE-2019-13232: Info-ZIP UnZip 6.0 mishandles the overlapping of files inside a ZIP container, leading to denial of service (resource consumption), aka a "better zip bomb" issue.

For the ClamAV v0.101.3 release notes, see the ClamAV blog.

Known Issue:

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.39

Release Date: December 10, 2018

Features

New features and changes in this release:

- The following information about scans is written to `/var/vcap/sys/log/clamav/clamdscan.log`
  - When a scan starts, a “Starting scheduled scan” message is output to the file.
  - When a scan ends, a “Scan Summary” message is output to the file.

For more information, see Monitoring ClamAV Logs.

Known Issue:

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.38

Release Date: October 29, 2018

Features

New features and changes in this release:

- The bundled ClamAV open source distribution is now v0.100.2.

Security Fixes

This release includes the following security fixes:

- CVE-2018-15378: A vulnerability in ClamAV’s MEW unpacking feature in v0.100.1 and earlier could allow a denial-of-service (DoS) condition.
- CVE-2018-14680: An issue in mspack/chmd.c in libmspack before 0.7alpha prevented it rejecting blank CHM filenames.
- CVE-2018-14681: Bad KWAJ file header extensions could cause a one- or two-byte overwrite.

- CVE-2018-14682: An off-by-one error in the TOLOWER() macro for CHM decompression. Additionally, v0.100.2 reverted v0.100.1's patch for CVE-2018-14679 and applied this newer fix instead.

For the ClamAV v0.100.2 release notes, see the ClamAV blog.

Known Issue:

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.36

Release Date: October 12, 2018

Features

New feature in this release:

- The ability to add signature names to an allowlist. For more information, see Add False Positives to an Allowlist.

Known Issue:

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.34

Release Date: September 4, 2018

Features

New feature in this release:

- The ability to specify a schedule for scan start times for daily scans

Known Issue

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.29

Release Date: July 30, 2018

Features

Changes in this release:

- The bundled ClamAV open source distribution is now v0.100.1.

Security Fixes

This release includes the following security fixes:

- CVE-2017-16932: parser.c in libxml2 before 2.9.5 does not prevent infinite recursion in parameter entities.
- CVE-2018-0360: ClamAV before 0.100.1 has an HWP integer overflow with a resultant infinite loop through a crafted Hangul Word Processor file.
- CVE-2018-0361: ClamAV before 0.100.1 lacks a PDF object length check, resulting in an unreasonably long time to parse a relatively small file.

For the ClamAV v0.100.1 release notes, see the ClamAV blog.

Known Issue

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.28

Release Date: June 28, 2018

Features

New features and changes in this release:

- Upgraded the bundled ClamAV open source distribution to v0.100.
- Added support for Canonical Ubuntu 16.04 (Xenial) stemcells.
- Removed developer acceptance tests from the bosh release package.
- Added support for cgroup configuration of hard CPU limit.

Resolved Issues

This release has the following fix:

- Fixed an issue with the shell script interpreter directive (“shebang”) that was specified in the job template for scheduled-scan. Previous releases used the interpreter directive `/bin/sh`, which prevented proper cgroup restriction. This has been corrected to `/bin/bash`.

Known Issue

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.5

Release Date: March 26, 2018

Features

New features and changes in this release:

- Upgrades the bundled ClamAV open source distribution to v0.99.4

Security Fixes

This release includes the following security fixes:

- CVE-2012-6706
- CVE-2017-6419
- CVE-2017-11423
- CVE-2018-1000085
- CVE-2018-0202

Compatibility fixes

- GCC 6
- C++11
- Updates golang to v1.10 for Linux builds

Known Issue

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

v1.4.1

Release Date: February 5, 2018

Features

New features and changes in this release:

- Upgrades the bundled ClamAV open source distribution to v0.99.3.

Security Fixes

This release includes the following security fixes:

- CVE-2017-12374
- CVE-2017-12375
- CVE-2017-12376
- CVE-2017-12377

- CVE-2017-12378
- CVE-2017-12379
- CVE-2017-12380

Known Issues

This release has the following issue:

- The clamd app forwards duplicate logs. This is because clamd controller outputs logs to both `/var/vcap/sys/log/monit/clamd.log` and `/var/vcap/sys/log/antivirus/clamd.log`.

View Release Notes for Another Version

To view the release notes for another product version, select the version from the dropdown at the top of this page.

Installing ClamAV Add-on for PCF

This topic describes how to install the ClamAV Add-on for Pivotal Cloud Foundry (PCF).

Prerequisites

To install ClamAV Add-on, you need:

- Named runtime configs

  If you have not already split your runtime config into multiple named files, do so before installing or upgrading the ClamAV Add-on for PCF. For general information about named runtime config files, see Configs.

- **A PCF operator with admin rights**: For more information, see Operators in the Pivotal Cloud Foundry documentation.
- Operations Manager (Ops Manager)
- At least 1 GB of memory free on each VM before deploying ClamAV Add-on. ClamAV Add-on uses 610 MB of memory.
- A local mirror to get ClamAV virus updates. For instructions to set up a local mirror, see Private Local Mirrors in the ClamAV documentation.

Note: Release version `x.x.x` in the `clamav.yml` samples below is arbitrary. Replace it with the version of ClamAV release downloaded from Pivotal Network.

Create the ClamAV Manifest

The ClamAV manifest is a YAML file that contains runtime config information for ClamAV Add-on. To create the ClamAV manifest for your deployment, follow the steps below:

clamav.yml Template for Linux

1. Create a file named `clamav.yml`, using the following code as a template.

```
releases:
- name: clamav
  version: x.x.x
addons:
- name: clamav
  jobs:
  - name: clamav
    release: clamav
```

```
    properties:
      clamav:
        database_mirror: 192.0.2.1
  include:
    stemcell:
    - os: ubuntu-trusty
    - os: ubuntu-xenial
```

2. (Optional) Set the properties `cpu_limit`, `enforce_cpu_limit` and `memory_limit`. To use these properties, place the following text under the `clamav` subsection of `clamav.yml`, as shown below:

```
...
properties:
  clamav:
    cpu_limit: VALUE-OF-CPU-LIMIT
    enforce_cpu_limit: TRUE|FALSE
    memory_limit: VALUE-OF-MEMORY-LIMIT-IN-BYTES
...
```

cpu_limit:

- Limits ClamAV to a percentage of available CPU resources when other processes are using CPU resources. Usage may exceed the limit if enough idle CPU cycles are available.
- Set to a whole number less than 100. For example, set to `50` to limit ClamAV to 50% CPU usage when other tasks are running.
- The default value is `10`.

enforce_cpu_limit:

- When `true`, the limit set by `cpu_limit` is always enforced.
- When `false`, the limit set by `cpu_limit` is only enforced when other processes are using CPU resources. Usage may exceed the limit if enough idle CPU cycles are available.
- This property is `false` by default.

Warning: If `enforce_cpu_limit` is set `true`, ensure `cpu_limit` is set high enough for ClamAV to execute normally. If the limit is too strict, ClamAV fails to start. For example, on Google Cloud Platform (GCP), an n1-standard-1 VM requires a CPU limit of more than 45%.

memory_limit:

- Limits the maximum amount of user memory (including file cache) in bytes used by ClamAV.
- The default value is `1073741824`.

clamav.yml Template for Windows

Create a file named `clamav.yml` using the following code as a template.

```
releases:
- name: clamav
  version: x.x.x
addons:
- name: clamav-windows
  jobs:
  - name: clamav-windows
    release: clamav
    properties:
      clamav:
        database_mirror: 192.0.2.1
  include:
    stemcell:
    - os: windows2019
    - os: windows2016
    - os: windows1803
```

General clamav.yml Template Configuration

1. In the `database_mirror` field of the template, provide the hostname or IP address of a private ClamAV update mirror. Environments that cannot connect to the internet should use an update mirror. If you do not specify a value, ClamAV defaults to an S3-based mirror for updates. For compliance reasons, only use the S3-based mirror in non-production environments.

   For instructions about setting up a local mirror, see Private Local Mirrors in the ClamAV documentation.

2. (Optional) If you have to use a proxy server to connect to the internet, add the `proxy_host` and `proxy_port` properties to your manifest.

   If the proxy server needs authentication, add the `proxy_user` and `proxy_password` properties, as shown below:

   ```
   releases:
   - name: clamav
     version: x.x.x
   addons:
   - name: clamav
     jobs:
     - name: clamav
       release: clamav
     properties:
       clamav:
         proxy_host: PROXY.LOCALDOMAIN
         proxy_port: PROXY-PORT
         proxy_user: PROXY-USERNAME
         proxy_password: PROXY-SECRET
   ...
   ```

Download ClamAV Add-on

To download the ClamAV Add-on software binary file and move it to your Ops Manager virtual machine, perform the steps below. If you intend to run the ClamAV Add-on on a PCF deployment that includes services or components that use Ubuntu Xenial stemcells, you should download ClamAV Add-on v1.4.28 or later.

Warning: Ensure that you are using named runtime configs. For more information, see Prerequisites above.

1. Download the ClamAV Add-on software binary from the Pivotal Network to your local machine.
2. To copy the binary to your Ops Manager VM, run the following command:

   ```
   scp -i PATH-TO-PRIVATE-KEY clamav-release.tar.gz ubuntu@YOUR-OPS-MANAGER-VM-IP:
   ```

   For example:

   ```
   $ scp -i ~/.ssh/my-key.pem clamav-1.3.28.tar.gz ubuntu@192.168.0.2:
   ```

3. To copy the ClamAV manifest, `clamav.yml` file, to your Ops Manager instance, run the following command:

   ```
   scp -i PATH-TO-PRIVATE-KEY clamav.yml ubuntu@YOUR-OPS-MANAGER-VM-IP:
   ```

   For example:

   ```
   $ scp -i ~/.ssh/my-key.pem clamav.yml ubuntu@192.168.0.2:
   ```

4. To SSH into Ops Manager, run the following command:

   ```
   ssh -i PATH-TO-PRIVATE-KEY ubuntu@YOUR-OPS-MANAGER-VM-IP
   ```

   For example:

   ```
   $ ssh -i ~/.ssh/my-key.pem ubuntu@192.168.0.2
   ```

5. To navigate to the location of the binary on the Ops Manager VM, run the following command:

   ```
   cd PATH-TO-BINARY
   ```

   For example:

   ```
   $ cd ~/clamav-1.3.28.tar.gz
   ```

Deploy the ClamAV Add-on

Perform the following steps to deploy the ClamAV Add-on:

1. Log in to the BOSH Director.
    1. On the Ops Manager VM, create an alias in the BOSH CLI for your BOSH Director IP

address. For example:

```
$ bosh alias-env my-env -e 10.0.0.3
```

1. Log in to the BOSH Director, specifying the newly created alias. For example:

```
$ bosh -e my-env log-in
```

2. Upload your release, specifying the path to the tarballed ClamAV binary, by running the following command:

```
$ bosh -e my-env upload-release ~/clamav-1.3.28.tar.gz
```

3. List the releases by running the following command, and confirm that ClamAV appears:

```
$ bosh -e my-env releases
```

4. Update your runtime configuration to include the ClamAV Add-on, specifying the path to the `clamav.yml` file you created above, by running the following command:

Note: If you installed other BOSH add-ons, you must merge the ClamAV manifest into your existing add-on manifest. Append the contents of `clamav.yml` to your existing add-on YML file.

```
$ bosh -e my-env update-runtime-config --name=clamav ~/clamav.yml
```

5. Verify that your runtime configuration changes match what you specified in the ClamAV manifest by running the following command:

```
$ bosh -e my-env runtime-config --name=clamav
```

For example:

```
$ bosh -e my-env runtime-config --name=clamav
Acting as user 'admin' on 'micro'
releases:
- name: clamav
   version: 1.3.28
addons:
  name: clamavv
    jobs:
      - name: clamav
        release: clamav
...
```

6. Navigate to your Installation Dashboard in Ops Manager.
7. If you are using Ops Manager v2.3 or later, click **Review Pending Changes**. For more information about this Ops Manager page, see Reviewing Pending Product Changes.
8. Click **Apply Changes**.

Configure Forwarding for ClamAV Alerts

The ClamAV BOSH release writes all alerts to the syslogs of the VMs in your deployment. You can use syslog forwarding to forward the alerts to a syslog aggregator.

Follow the steps to Configure System Logging in the Pivotal Application Service (PAS) tile. The syslog aggregator that you specify receives all alerts generated on PAS VMs, including the ClamAV alerts.

Note: When you configure syslog forwarding, ensure there is enough disk space for the logs. Make sure that log rotation is frequent enough. If in doubt, rotate the logs hourly or when they reach a certain size. Pivotal recommends forwarding logs to a remote syslog aggregation system.

Verify Your ClamAV Add-on Installation

1. BOSH SSH into one of the VMs in your deployment.
2. Run `monit summary`. Look for the following processes in the output:

```
The Monit daemon 5.2.4 uptime: 3d 0h 56m
Process 'clamd'                     running

Process 'freshclam'                 running
```

3. If `monit summary` does not list `clamd` and `freshclam`, do the following:
    1. Try to start the ClamAV processes by running the following commands:

    ```
    $ monit start clamd
    $ monit start freshclam
    ```

    2. Run `monit summary` again. If you do not see the processes mentioned above, check `/var/vcap/sys/log/clamav` logs for errors.
4. If `monit summary` does list `freshclam` and `clamd`, create a file in `/var/vcap` on the VM with the following contents:

    ```
    X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
    ```

    This is a virus signature used to test anti-virus software.

After `clamdscan` completes, a notification should appear in `/var/log/syslog`. This scan can take up to an hour.

ClamAV Log Format

See Monitoring ClamAV Logs

Add False Positives to an Allowlist

Note: Allowlist is available in ClamAV Add-on v1.4.36 and later.

If a scan is reporting false positives, report the issue to ClamAV. For more information about false positives, see ClamAV Reports False Positives.

It takes about a week for ClamAV to verify and publish a new database. If a week is too long, the ClamAV release provides the option to add the signature names to an allowlist.

To add signature names to an allowlist, do the following:

1. Add the `whitelist` property and append the false positive signature names. For example:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
  jobs:
  - name: clamav
    release: clamav
    properties:
      clamav:
        <strong>whitelist:</strong>
        <strong>  - Eicar-Test-Signature</strong>
        <strong>  - Clamav.Test.File-7</strong>
        <strong>  - Win.Test.EICAR_NDB-1</strong>
        <strong>  - Eicar-Signature</strong>
...
```

2. Apply changes by updating your runtime config. For more information, see steps 4 to 7 of Deploy the ClamAV Add-on.

Enable On-Access Scan

Note: On-Access Scan is not supported on Windows.

ClamAV offers immediate file scanning upon file modification. This feature may reduce the time it takes to detect and report malware.

To enable on-access file scanning with the `on_access` runtime config property:

1. In the `clamav.yml` file, add the `on_access` property under the `clamav` property, and set `on_access` to `yes`, as shown in bold:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
  jobs:
  - name: clamav
    release: clamav
    properties:
      clamav:
        on_access: yes
        scheduled: yes
...
```

2. Apply changes by updating your runtime config. For more information, see steps 4 to 7 of Deploy the ClamAV Add-on above.

Configure Scheduled Scan

ClamAV can be configured to run a virus scan hourly or daily, with daily scan being the default.

To change the scheduled scan to run hourly:

1. In the `clamav.yml` file, add the property `schedule_interval` under the `clamav` property, and set it to `hourly`, as shown below:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
  jobs:
  - name: clamav
    release: clamav
    properties:
      clamav:
        scheduled: yesv
        schedule_interval: hourly
...
```

2. Apply changes by updating your runtime config. For more information, see steps 4 to 7 of Deploy the ClamAV Add-on above.

To restrict the time interval when a daily scan runs, do the following:

1. For ClamAV Add-on v1.4.34 and later, restrict the timeframe when the scheduled scan can run by specifying a time range in 24-hour format in the properties `first_scheduled_scan_time` and `last_scheduled_scan_time`, as shown below:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
  jobs:
  - name: clamav
    release: clamav
    properties:
      clamav:
        scheduled: yes
        schedule_interval: daily
        first_scheduled_scan_time: "2:00"
    last_scheduled_scan_time: "4:30"
...
```

2. Apply changes by updating your runtime config. For more information, see steps 4 to 7 of Deploy the ClamAV Add-on above.

Deactivate Scheduled Scan

To deactivate the scheduled scan, do the following:

1. In the `clamav.yml` file, set the property `scheduled` to `no`.
2. Apply changes by updating your runtime config. For more information, see steps 4 to 7 of Deploy the ClamAV Add-on above.

Choose the Action on Infected Files

You can configure ClamAV to take action when infected files are found. By default, a notification is sent to the syslog when an infected file is found. However, you can specify other actions, as described Step 2 below.

1. In the `clamav.yml` file, add the `action` property under the `clamav` property and, optionally, the `action_destination` property, as shown below:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
  jobs:
  - name: clamav
    release: clamav
    properties:
      clamav:
        action: ACTION
        action_destination: PATH
...
```

2. Replace `ACTION` with one of the following values:
   - `notify`: The default, only send a notification to syslog
   - `remove`: Delete the infected file from the filesystem
   - `move`: Move the infected file to the directory location specified by `action_destination`
   - `copy`: Copy the infected file to the directory location specified by `action_destination`

   If you don't supply an action, the function fails.
3. Replace `PATH` with the directory location where you want the infected files moved or copied to. The system does not scan the moved-to or copied-to location. If the directory path is not valid, the function fails.

> ⚠ **Warning:** If `action` is `move` or `copy`, make sure the `action_destination` path is also added to the `exclude_paths`.
> If `action_destination` is not listed under `exclude_paths`, ClamAV:
> - Detects the moved or copied file
> - Logs redundant alerts

- Creates additional copies of the detected file

Example configuration:

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
    jobs:
    - name: clamav
      release: clamav
      properties:
        clamav:
          action: move
          action_destination: /var/vcap/data/clamav/found
          exclude_paths:
          - ^/proc/
          - ^/sys/
          - ^/var/vcap/data/clamav/found/
...
```

Exclude Files and Directories

You can configure ClamAV to exclude files and directories from being scanned. By default, the scan excludes `/proc` and `/sys` directories.

To exclude files and directories from the scan, do the following:

1. Add the `exclude_paths` property and append regex matching file paths to the list, as shown below:

Note: Pivotal recommends that you include the `/proc` and `/sys` directories on the `exclude_paths` list. This is because they must be on the list to remain excluded.

```
releases:
- name: clamav
  version: x.x.x
addons:
  name: clamav
    jobs:
    - name: clamav
      release: clamav
      properties:
        clamav:
          exclude_paths:
          - ^/proc/
          - ^/sys/
          - ADDITONAL-EXCLUSIONS
...
```

2. Apply changes by updating your runtime config. For instructions, see steps 4 to 7 of Deploy the ClamAV Add-on above.

(Optional) Exclude Duplicate Logs on Garden Containers

When ClamAV scan results detect potential malware on the Garden containers, logs are reported for both the `diff` and `rootfs` directories.

This is because the `rootfs` directory is the projection of the `diff` directory on top of a base image layer. Therefore, it is safe to ignore the `rootfs` directory. GrootFS mounts the underlying volumes using OverlayFS to a point in the images directory. This mount point is the `rootfs` directory for the container and is read-write.

For more information about GrootFS OverlayFS implementation, see Volumes in the Cloud Foundry documentation.

To configure ClamAV to ignore duplicate logs for these directories, do the following:

1. In Exclude Files and Directories above, add the following pattern to the `exclude_paths` property:

```
clamav:
  exclude_paths:
    ...
    # When scanning Garden containers,
    # ignore duplicate log messages from .../UUID/rootfs/
    - ^/var/vcap/data/grootfs/store/(un)?privileged/images/[\w-]+/rootfs/.*$
```

Note: Adding this ignore pattern means that files and directories in the `/var/vcap/data/grootfs/store/unprivileged/images/UUID/rootfs` and `/var/vcap/data/grootfs/store/privileged/images/UUID/rootfs` directories are ignored by ClamAV. `UUID` is the ID of the container.

For an example log message, see Container Log Messages.

(Optional) Exclude Duplicate Logs on Containers in Enterprise PKS

When ClamAV scan results detect potential malware on containers of the Kubernetes worker node VMs in Enterprise PKS, logs are reported for both the `diff` and `merged` directories.

This is because the `merged` directory is the projection of the `diff` directory on top of a base image layer. Therefore, it is safe to ignore the `merged` directory.

For more information about Docker OverlayFS implementation, see Use the OverlayFS storage driver in the Docker documentation.

To configure ClamAV to ignore duplicate logs for these directories, do the following:

1. In Exclude Files and Directories above, add the following pattern to the `exclude_paths` property:

```
clamav:
  exclude_paths:
    ...
```

```
    # When scanning Kubernetes containers,
    # ignore duplicate log messages from .../UUID/merged/
    -  ^/var/vcap/store/docker/docker/overlay2/\w+/merged/.*$
```

Note: Adding this ignore pattern means that files and directories in the `/var/vcap/store/docker/docker/overlay2/UUID/merged` directory are ignored by ClamAV. `UUID` is the ID of the container.

For an example log message, see Container Log Messages.

Updating ClamAV Add-on for PCF to Run with PAS for Windows v2.3

Read this topic if you use Pivotal Application Service for Windows (PAWS).

This topic describes how to update your runtime config to add the new Windows stemcell version.

Breaking Change: If you upgrade to PASW v2.3, you must update your runtime config to include the new Windows stemcell. Otherwise, the PASW VMs will not be protected by ClamAV.

Windows Stemcell Renamed

The new Windows stemcell name indicates the *stemcell* version instead of the *Windows Server* version.

PASW version	Stemcell name	Stemcell version
2.2 and earlier	`windows2016`	1709
2.3 and later	`windows1803`	1803

For general information about stemcells and PASW, see PAS for Windows v2.3 Release Notes.

Add the windows1803 Stemcell to the ClamAV Add-on

To add the `windows1803` stemcell property to the runtime config, do the following.

1. SSH into the Ops Manager VM. For instructions, see SSH into Ops Manager.
2. To retrieve and save the ClamAV add-on runtime config, run the following command:

   ```
   bosh -e BOSH-ENVIRONMENT runtime-config –name clamav > /tmp/clamav.yml
   ```

   Where `BOSH-ENVIRONMENT` is the alias you set for the BOSH Director.

   For example:

   ```
   $ bosh2 -e my-env runtime-config –name clamav > /tmp/clamav.yml
   ```

3. Edit the `clamav.yml` file to add `- os: windows1803` under `stemcell` in the `clamav-windows` section, as shown below:

   ```
   addons:
   - name: clamav-windows
   ```

```
jobs:
- name: clamav-windows
  release: clamav
  properties:
    clamav:
      database_mirror: 192.0.2.1
include:
  stemcell:
  - os: windows2019
  - os: windows2016
  - os: windows1803
```

4. To update the runtime config, run the following command:

   ```
   bosh2 -e BOSH-ENVIRONMENT update-runtime-config --name=clamav /tmp/clamav.yml
   ```

   For example:

   ```
   bosh2 -e my-env update-runtime-config --name=clamav /tmp/clamav.yml
   ```

5. Navigate to the Installation Dashboard in Ops Manager and do the following to complete the installation:

   1. If you are using Ops Manager v2.3 or later, click **Review Pending Changes**. For more information about this Ops Manager page, see Reviewing Pending Product Changes.
   2. Click **Apply Changes**.

Updating ClamAV Add-on for PCF to Run with Xenial Stemcells

Pivotal Cloud Foundry (PCF) products and tiles that are released after July 2018 require Ubuntu Xenial stemcells instead of Ubuntu Trusty stemcells. You might have to modify your ClamAV Add-on for PCF deployment if you use PCF products running on Xenial.

This topic describes how to determine if your existing deployment of ClamAV Add-on can protect VMs that run on Xenial.

This topic also explains how to update your ClamAV Add-on if it does not support Xenial.

Follow the instructions on this page if you use ClamAV Add-on with any PCF products or tiles that use Xenial stemcells. See Product Tiles that Use Xenial Stemcells below.

Do I Need to Modify the ClamAV Add-on?

ClamAV Add-on v1.4.28 and later can run correctly on Xenial-based VMs if the ClamAV runtime config includes the `ubuntu-xenial` property.

Review the following table and make any required changes before you upgrade to Xenial stemcells.

If you are using this version of the ClamAV Add-on...	do the following...

1.4.29	Verify that your runtime config file, `clamav.yml`, includes:
	`stemcell:`
	`- os: ubuntu-trusty`
	`- os: ubuntu-xenial`
	If it does not, then follow the procedure, Add the Xenial Stemcell Property to ClamAV Add-on below.

v1.4.28	Follow the procedure, Add the Xenial Stemcell Property to the ClamAV Add-on below.
v1.4.5 or earlier	Install ClamAV Add-on v1.4.29.

If you use ClamAV Add-on without adding the `ubuntu-xenial` property to the runtime config, the VMs running on Xenial are not scanned for viruses.

If you add the `ubuntu-xenial` property but do not upgrade the ClamAV Add-on to v1.4.28 or later, then the ClamAV processes use excessive CPU.

Product Tiles that Use Xenial Stemcells

Ensure that you have added the `ubuntu-xenial` property to the ClamAV runtime config before you install any product tiles that use Xenial stemcells.

For a list of PCF tile releases that now use Xenial, see [Tiles Using Xenial Stemcells in PCF](https://docs.pivotal.io/stemcells/xenial-tiles.html.

Add the Xenial Stemcell Property to the ClamAV Add-on

If you use ClamAV Add-on v1.4.28 or later without the `ubuntu-xenial` property in the runtime config, then you must add it to your existing `clamav.yml` and redeploy.

Follow these steps:

1. SSH into the Ops Manager VM. For instructions, see SSH into Ops Manager.
2. To retrieve and save the ClamAV add-on runtime config, run the following command:

   ```
   bosh -e BOSH-ENVIRONMENT runtime-config -name clamav > /tmp/clamav.yml
   ```

   Where `BOSH-ENVIRONMENT` is the alias you set for the BOSH Director.

   For example:

   ```
   $ bosh2 -e my-env runtime-config -name clamav > /tmp/clamav.yml
   ```

3. Edit the `clamav.yml` file to add `- os: ubuntu-xenial` under `properties: {}` as shown below:

   ```
   addons:
   - name: clamav
     jobs:
     - name: clamav
       release: clamav
       properties: {}
     include:
       stemcell:
       - os: ubuntu-trusty
       - os: ubuntu-xenial
   ```

4. (Optional) If you plan to update to PAS for Windows v2.3, do Step 3 of Add the windows1803 Stemcell to the ClamAV Add-on.
5. To update the runtime config, run the following command:

   ```
   bosh2 -e BOSH-ENVIRONMENT update-runtime-config --name=clamav /tmp/clamav.yml
   ```

   For example:

   ```
   bosh2 -e my-env update-runtime-config --name=clamav /tmp/clamav.yml
   ```

6. Navigate to the Installation Dashboard in Ops Manager.
7. If you are using Ops Manager v2.3 or later, click **Review Pending Changes**. For more information about this Ops Manager page, see Reviewing Pending Product Changes.
8. Click **Apply Changes**.

Upgrading ClamAV Add-on for PCF

This topic describes how to upgrade the ClamAV Add-on for PCF.

For product versions and upgrade paths, see Upgrade Planner.

Compatibility and Prerequisites

See the following topics to ensure you have the required component versions and prerequisites:

- Product Snapshot
- Prerequisites

Upgrade ClamAV Add-on for PCF

To upgrade the ClamAV add-on to a later version, do the following:

1. Download the ClamAV add-on software binary from the Pivotal Network to your local machine.
2. To copy the software binary to your Ops Manager VM, run the following command:

   ```
   scp -i PATH-TO-PRIVATE-KEY clamav-VERSION.tar.gz ubuntu@YOUR-OPS-MANAGER-VM-IP:
   ```

   For example:

   ```
   $ scp -i ~/.ssh/my-key.pem ~/Downloads/clamav-1.4.34.tgz ubuntu@192.16
   8.0.2:
   ```

3. SSH into the Ops Manager VM. For instructions, see SSH into Ops Manager.
4. Retrieve the latest runtime config by running the following command:

   ```
   bosh -e BOSH-ENVIRONMENT runtime-config  --name clamav > PATH-TO-SAVE-THE-RUNTI
   ME-CONFIG
   ```

   For example:

   ```
   bosh -e my-env runtime-config  --name clamav > /tmp/clamav.yml
   ```

5. Upload the ClamAV release to BOSH. This is the release that you downloaded from Pivotal Network above.

   ```
   bosh -e BOSH-ENVIRONMENT upload-release PATH-TO-NEW-CLAMAV-RELEASE
   ```

   For example:

```
bosh -e my-env upload-release ~/clamav-1.4.34.tgz
```

6. Edit the clamav runtime config to set the new release version. This version should match the version you downloaded from Pivotal Network.

   For example, edit the version in `/tmp/clamav.yml` as follows:

   ```
   releases:
   - name: clamav, version: <strong>1.4.34</strong>
   ```

7. Update the runtime config:

   ```
   bosh -e BOSH-ENVIRONMENT update-runtime-config --name=clamav PATH-TO-SAVE-THE-R
   UNTIME-CONFIG
   ```

   For example:

   ```
   bosh -e my-env update-runtime-config --name=clamav /tmp/clamav.yml
   ```

8. Navigate to your Ops Manager **Installation Dashboard**.
9. If you are using Ops Manager v2.3 or later, click **Review Pending Changes**. For more information about this Ops Manager page, see Reviewing Pending Product Changes.
10. Click **Apply Changes**.

Monitoring ClamAV Logs

This topic contains sample logs emitted by ClamAV.

You can use these samples to configure a Security Information and Event Management (SIEM) system to verify regular activity and generate alerts for virus detections or outdated virus signatures.

ClamAV Logs

There are three distinct ClamAV apps that run on each VM, freshclam, clamd, and clamdscan. These apps work together to detect viruses and protect the VM.

Each app writes its own log file. You need to monitor each of these files to know if ClamAV Addon for PCF is working correctly and if viruses have been found.

Pivotal recommends that you enable syslog forwarding so that the messages from each of the three log files is aggregated into the syslog file on the remote syslog server. Then you can use your preferred monitoring and alerting tool to review the Clamav log messages.

For an example of how ClamAV messages appear in the syslog file, see Syslog Format below.

For information about each app, see freshclam, clamd, and clamdscan below.

freshclam App

The freshclam app updates the database that stores the known virus signatures.

The messages output by the freshclam app indicate when freshclam checks for updates, what the download progress is, and the downloaded signature version.

The log file for the freshclam app is `/var/vcap/sys/log/clamav/freshclam.log`.

clamd App

The Clam AntiVirus Daemon (clamd) listens for incoming connections on Unix or the TCP socket. clamd works with clamdscan to scan files or directories. The clamd job uses the database of virus signatures that the freshclam job updates.

The messages output by the clamd app show files where viruses are found, the name of the virus signature, and any action taken, such as moving, copying, or deleting.

The log file for the clamd app is `/var/vcap/sys/log/clamav/clamd.log`.

clamdscan App

The clamdscan app scans files and directories for viruses using the clamd daemon.

The messages output by the clamdscan app show when a clamdscan is initiated and writes a scan summary on completion.

The log file for the clamd app is `/var/vcap/sys/log/clamav/clamdscan.log`.

Log Messages

The following tables lists common messages that you see when ClamAV apps write to log files:

Message	App	Meanings	Healthy/ Unhealthy?
Check for Updates	freshclam	States that the freshclam app is checking the configured remote mirror for an update to the local virus signature database.	Healthy
Update the Virus Database	freshclam	States that the virus database is being updated.	Healthy
Cannot Download CLD Database Files	freshclam	States that freshclam could not download the latest uncompressed databases. These database files include the `main.cld`, `daily.cld`, and `bytecode.cld` files. They are optional for ClamAV to run.	Healthy
Virus Database Is Up-to-Date	freshclam	States that the virus database is up-to-date.	Healthy
Virus Database is Older Than 7 Days	freshclam	States that the virus database is stale. Based on configuration, freshclam checks hourly or daily.	Unhealthy
Process Terminated	freshclam	freshclam should only terminate during a deployment.	Unhealthy (Will be triggered by deployments)
Start clamd	clamd	States that a clamd daemon is starting.	Healthy

Check for Updated Virus Signatures	clamd	clamd checks if freshclam has updated the local virus signature database.	Healthy
Virus Detected	clamd	Gives the name and location of the virus that was found and the virus signature that it matches.	Unhealthy
Virus Removed	clamd	Gives the name of the virus file that was found and states that the file was deleted.	Unhealthy
Virus Moved	clamd	Gives the name of the virus file found and where it was moved to. The virus file is deleted from original location.	Unhealthy
Virus Copied	clamd	Gives the name of the virus file found and where it was copied to. The virus file remains at original location.	Unhealthy
Process Terminated	clamd	Both clamd and freshclam should always be running. If the process was terminated, meaning the clamd daemon has stopped, then this error appears and can indicate a problem. Neither on-access scanning nor scheduled scanning is possible if the process state is terminated.	Unhealthy (Will be triggered by deployments)
Start Scheduled Scan	clamdscan	States when the scan starts. Use the time stamp on the message to determine this.	Healthy
Scan Finished	clamdscan	Gives time elapsed for scan and how many infected files were found.	Healthy

freshclam Log Messages

The freshclam job on each VM is responsible for updating the database that stores the known virus signatures.

freshclam log entries relate to whether or not the virus-signature database is up-to-date.

- Check for Updates

```
ClamAV update process started at Wed Nov 28 15:58:23 2018
```

- Update the Virus Database

```
Downloading main.cvd [100%]
main.cvd updated (version: 58, sigs: 4566249, f-level: 60, builder: si
gmgr)
Downloading daily.cvd [100%]
daily.cvd updated (version: 25135, sigs: 2155329, f-level: 63, builder
: neo)
Downloading bytecode.cvd [100%]
bytecode.cvd updated (version: 327, sigs: 91, f-level: 63, builder: ne
o)
Database updated (6721669 signatures) from pivotal-clamav-mirror.s3.am
azonaws.com (IP: 52.216.169.19)
```

- Virus Database Is Up-to-Date

```
main.cvd is up to date (version: 58, sigs: 4566249, f-level: 60, build
er: sigmgr)
daily.cvd is up to date (version: 25135, sigs: 2155329, f-level: 63, b
uilder: neo)
bytecode.cvd is up to date (version: 327, sigs: 91, f-level: 63, build
er: neo)
```

- Cannot Download CLD Database Files

```
WARNING: getfile: Unknown response from pivotal-clamav-mirror.s3.amazo
naws.com (IP: 52.216.233.147): HTTP/1.1 403
WARNING: Can't download main.cld from pivotal-clamav-mirror.s3.amazona
ws.com
WARNING: getfile: Unknown response from pivotal-clamav-mirror.s3.amazo
naws.com (IP: 52.216.233.147): HTTP/1.1 403
WARNING: Can't download daily.cld from pivotal-clamav-mirror.s3.amazon
aws.com
WARNING: getfile: Unknown response from pivotal-clamav-mirror.s3.amazo
naws.com (IP: 52.216.233.147): HTTP/1.1 403
WARNING: Can't download bytecode.cld from pivotal-clamav-mirror.s3.ama
zonaws.com
```

- Virus Database is Older Than 7 Days

```
[LibClamAV] ************************************************
[LibClamAV] ***  The virus database is older than 7 days!  ***
[LibClamAV] ***   Please update it as soon as possible.    ***
[LibClamAV] ************************************************
```

- Process Terminated

```
Update process terminated
```

clamd Log Messages

clamd is the anti-virus scanner that searches for viruses. The clamd job uses the database of virus signatures that the freshclam job updates.

- Start clamd

```
Wed Nov 28 15:58:47 2018 -> +++ Started at Wed Nov 28 15:58:47 2018
Wed Nov 28 15:59:02 2018 -> Self checking every 600 seconds.
```

- Check for Updated Virus Signatures

```
SelfCheck: Database status OK.
```

```
SelfCheck: Database modification detected. Forcing reload
```

```
No stats for Database check - forcing reload
```

- Virus Detected

```
/var/vcap/data/test.txt: Eicar-Test-Signature FOUND
```

- Virus Removed

```
/var/vcap/data/test.txt: Removed.
```

- Virus Moved

```
/var/vcap/data/test.txt: moved to '/var/vcap/data/clamav/found/test.tx
t.001'
```

- Virus Copied

```
/var/vcap/data/test.txt: copied to '/var/vcap/data/clamav/found/test.t
xt.001'
```

- Process Terminated

```
Wed Nov 28 19:25:23 2018 -> Pid file removed.
Wed Nov 28 19:25:23 2018 -> --- Stopped at Wed Nov 28 19:25:23 2018
Wed Nov 28 19:25:23 2018 -> Socket file removed.
```

clamdscan Log Messages

clamdscan searches files and directories for viruses.

- Start Scheduled Scan

  This is not provided in ClamAV Add-on for PCF v1.4.38 and earlier.

```
Starting scheduled scan
```

- Scan Finished

  This is not provided in ClamAV Add-on for PCF v1.4.38 and earlier.

Sample:

```
----------- SCAN SUMMARY -----------
Infected files: 1
Time: 346.887 sec (5 m 46 s)
```

Container Log Messages

Examples of ClamAV log messages from Garden containers and Docker containers are as follows:

- For a Garden container in Pivotal Application Service (PAS)

```
/var/vcap/data/grootfs/store/unprivileged/images/2264d474-3e57-4934-50
4f-ddbb/diff/home/vcap/app/public/test.html:
Eicar-Test-Signature FOUND
```

- For a Docker container in Enterprise Pivotal Container Service (Enterprise PKS)

```
/var/vcap/store/docker/docker/overlay2/53322c6f7c25bb00224bb03cdfc285e
141471d746d5c7a8c5a65db56fda56ecb/diff/test.html:
Eicar-Test-Signature FOUND
```

ClamAV Log Format

The logs that ClamAV itself outputs do not adhere to a specific structure. However, the syslog forwarder component, which is on all VMs, encapsulates ClamAV's log and prepends the necessary headers so that the resulting logs adhere to the syslog format.

With syslog-forwarder, the syslog format is:

```
<PRI> \
VERSION \
TIMESTAMP \
HOST \
APP-NAME \
PROC-ID \
MSG-ID \
[instance@47450 \
director="DIRECTOR" \
deployment="DEPLOYMENT" \
group="INSTANCE-GROUP" \
az="AVAILABILITY-ZONE" \
id="ID"] \
MESSAGE \
```

Where:

- `<PRI>` is `<14>`.
- `APP-NAME` is `freshclam`, `clamdscan`, or `clamd`.
- `MESSAGE` is the output from a ClamAV app. Examples of the output messages are shown in Log Messages above.

For example, the first two lines of the "Scan Finished" message appearing in the syslog file below:

```
<14> \
1 \
2018-12-07T21:48:02.119539Z \
10.0.0.3 \
clamav \
rs2 \
- \
[instance@12345 \
director="" \
deployment="clamav-trusty-aaaa-80" \
group="clamav" \
az="z1" \
id="abcdef01-8901-42a5-ad58-8b4c1a2de881"] \
----------- SCAN SUMMARY -----------
<14> \
1 \
2018-12-07T21:48:02.11954Z \
10.0.0.3 \
clamav \
rs2 \
- \
[instance@12345 \
director="" \
deployment="clamav-trusty-rlee-80" \
group="clamav" \
az="z1" \
id="abcdef01-8901-42a5-ad58-8b4c1a2de881"] \
Infected files: 0
```

For more information, see Format in the syslog-release GitHub repository.

Troubleshooting ClamAV Add-on for PCF

This topic provides instructions for troubleshooting the ClamAV Add-on for PCF and verifying that it is protecting your Pivotal Cloud Foundry (PCF) deployment.

ClamAV Installation Issues

Ops Manager Fails to Apply Changes

Symptom

Applying changes in Ops Manager fails. The bottom of the changelog contains an error message similar to the following:

```
Started updating job nats > nats/0 (12bfae02-b4af-4104-b2bd-227ff07b2d92) (c
anary). Done (00:02:31)
  Failed updating job etcd_server > etcd_server/0 (f8e492bf-db09-4d38-8a73-5
cf69d7b8a11) (canary): 'etcd_server/0 (f8e492bf-db09-4d38-8a73-5cf69d7b8a11)
' is not running after update. Review logs for failed jobs: clamd (00:05:53)

Error 400007: 'etcd_server/0 (f8e492bf-db09-4d38-8a73-5cf69d7b8a11)' is not
running after update. Review logs for failed jobs: clamd
```

Explanation

The ClamAV mirror server was unavailable during initial deployment.

Solution

Review the manifest file, and replace the `database_mirror` key with the address of a stable mirror server. If you do not have a stable mirror server for reliable initial deployment, use the S3-based mirror: `pivotal-clamav-mirror.s3.amazonaws.com`

Issues

ClamAV Is Not Detecting Malware

Symptom

Malware signature or sample malware is not detected, even though the ClamAV daemon is properly configured.

Explanation

Virus signatures are not up-to-date.

Solution

To resolve this issue, verify that:

- The configuration checks have been completed.
- The mirror server is correctly configured.
- The mirror server is available on the network from within the PCF private subnet.
- At least one hour has elapsed. One hour is the default scan schedule interval.

If the local mirror is up-to-date and ClamAV still fails to detect a malware sample, you might have encountered a new threat. Pivotal recommends alerting the community using existing channels and reporting the suspicious file directly to the ClamAV team.

Note: Pivotal does not provide support for ClamAV detection failures, mirror coordination, or threat tracking activity.

ClamAV Reports False Positives

Symptom

ClamAV reports a false positive result such as a non-malicious file reported to be a virus.

Explanation

ClamAV compares files to its database of known malicious patterns. ClavAV might detect a non-malicious file as a virus due to a coincidental similarity to those patterns.

Solution

Submit false positive reports to ClamAV. You can also subscribe to the ClamAV email list to be kept up-to-date with ClamAV database changes. It takes about a week for ClamAV to verify and publish a new database.

CPU Spikes While Using ClamAV

Symptom

ClamAV is taking more CPU resources than assigned in its configuration.

Explanation

ClamAV resource consumption is restricted using cgroups. ClamAV is resource-limited whenever other processes are active. However, cgroups enables ClamAV to occupy more CPU resources

when all other processes are idle, because it does not impact their performance.

Solution

This is expected behavior from cgroups. If a hard limit is required, configure the `enforce_cpu_limit` Linux property. For more information, see clamav.yml Template for Linux Linux property.

Out of Memory While Using ClamAV

Symptom

ClamAV fails to start and `/var/log/syslog` reports `Memory cgroup out of memory: Kill process` on the `clamd` process similar to the following:

```
2019-02-20T19:35:40.249205+00:00 localhost kernel: [  254.669948] Memory cgr
oup out of memory: Kill process 7493 (clamd) score 586 or sacrifice child
2019-02-20T19:35:40.249205+00:00 localhost kernel: [  254.679053] Killed pro
cess 7527 (clamd) total-vm:786136kB, anon-rss:626692kB, file-rss:1592kB
```

Explanation

ClamAV resource consumption is restricted by cgroups. The `clamd` process is terminated if the memory usage limit is exceeded. When memory swapping is disabled by other BOSH jobs, the ClamAV resource requires a larger memory limit.

Solution

This is expected behavior from cgroups. To configure the memory limit, configure the `memory_limit` Linux property. For more information, see clamav.yml Template for Linux.

Warning: When updating the memory limit, ensure that all VMs, including errand VMs, have sufficient memory resources.

Insufficient CPU Limit While Using ClamAV

Symptom

ClamAV fails to start during deployment. However, the `clamd` and `freshclam` processes eventually run.

The deployment failure log looks similar to the following:

```
Task 1071 | 19:40:49 | Updating instance clamav_1: clamav_1/d5cfe4bd-b606-43
72-8481-187f4cf57e6c (0) (canary) (00:05:26)
                 L Error: 'clamav_1/d5cfe4bd-b606-4372-8481-187f4cf57e6c (0)'
 is not running after update. Review logs for failed jobs: clamd, freshclam
```

When you run `bosh -d DEPLOYMENT instances --ps`, you see that the the `clamd` and `freshclam`

processes are running successfully after the failed deployment.

For example:

```
$ bosh -d clamav_1/d5cfe4bd-b606-4372-8481-187f4cf57e6c instances --ps

Instance                                        Process    Process State  AZ
 IPs
clamav_1/d5cfe4bd-b606-4372-8481-187f4cf57e6c  -          running        z1
 10.0.0.7
~                                               clamd      running        -
 -
~                                               freshclam  running        -
 -
```

Explanation

ClamAV startup is CPU intensive and, if restricted, can prevent ClamAV from starting up correctly.

Solutions

- Ensure `cpu_limit` is set high enough for ClamAV to execute normally. If the limit is too strict, ClamAV fails to start. For example, an n1-standard VM on GCP requires `cpu_limit` to be greater than 45. For more information, see clamav.yml Template for Linux.
- Set `enforce_cpu_limit` to `false`. This allocates more CPU cycles to ClamAV if other processes are not using CPU resources.
  For more information, see clamav.yml Template for Linux.
- From the **Ops Manager Installation Dashboard**, navigate to the tile with the failing `clamav_1` job. On the **Resource Config** pane, adjust the **VM Type** for the **ClamAV** job to have sufficient CPU resources.

Uninstalling the ClamAV Add-on for PCF

This topic describes how to uninstall the ClamAV Add-on for PCF from your deployment, and how to verify the uninstallation.

Uninstall ClamAV Add-on

To uninstall the ClamAV Add-on, follow the steps below:

1. Retrieve the latest runtime config by running the following command:

   ```
   bosh -e my-env runtime-config > PATH-TO-SAVE-THE-RUNTIME-CONFIG
   ```

   Where `PATH-TO-SAVE-THE-RUNTIME-CONFIG` is the location that you want to save the runtime configuration.

   For example:

   ```
   $ bosh -e my-env runtime-config > /runtime/config/
   ```

2. In the runtime config, remove all ClamAV properties under the `releases:` and `addons:` sections.

3. Update the runtime config.

   ```
   bosh -e my-env update-runtime-config --name=clamav PATH-TO-SAVE-THE-RUNTIME-CON
   FIG
   ```

   Where `PATH-TO-SAVE-THE-RUNTIME-CONFIG` is the location of the runtime configuration you are updating.

   For example:

   ```
   $ bosh -e my-env update-runtime-config --name=clamav /runtime/config/
   ```

4. Navigate to your **Installation Dashboard** in Ops Manager.

5. If you are using Ops Manager v2.3 or later, click **Review Pending Changes**. For more information about this Ops Manager page, see Reviewing Pending Product Changes.

6. Click **Apply Changes**.

7. Wait for the installation to complete.

Verify the Uninstallation

To verify the uninstallation of the ClamAV Add-on is successful, follow the steps below:

1. Use `bosh ssh` to SSH into one of the VMs in your deployment. For more information, see BOSH SSH
2. Run `monit summary`. If ClamAV has uninstalled successfully, it does not show the `clamd` or `freshclam` processes.