USING APPLICARE › APPLICATIONS

Alerts

Applicare 9.2 · Updated August 29, 2024 · alerts · **Pending technical review**

User Role and Alerts

Super Admin, Users with Admin rights or App Admin rights for an application can add/view alerts related to that application. User with read only access for an application can only view predefined alerts on the application.

Applicare automatically generate platform alerts for CPU, disk and memory utilization for the agents if it is configured in MySQL or Oracle database.

Configure Alerts

Applicare allows users to set alerts and receive email notifications based on custom conditions. It offers the following types of alerts:

Platform Alerts	**- Operating System and JVM metrics (CPU, Memory, Swap, Heap Usage)**
JMX Alerts	- Set up alerts based on MBean attributes
SNMP Alerts	- Create alerts for values on SNMP capable devices
Tuxedo Queue Alerts	- Set up alerts for Tuxedo Queue
Tuxedo Service Alerts	- Create alerts on Tuxedo Services
UEM Slow Transactions Alerts	- Create User Experience alerts for slow transactions
Slow Transactions	- Create alerts on slow transactions as detected by IntelliSense
Business Transactions - SLA Avg	- Create alerts on Business Transactions exceeding SLA Avg
Business Transactions - SLA Max	- Create alerts on Business Transactions exceeding SLA Max

Tools -> Manage Alerts enable you to create, edit, pause or activate alerts. It also allows you to assign a command to be executed when an alert is fired. Applicare allows any shell/command script execution on any location where Applicare agent is running. This is very valuable for doing diagnosis or taking remediation actions when a certain alert is generated.

For creating high CPU utilization alert select New and platform alert

> [Screenshot 1 could not be downloaded during import — see migration report]

Click on the browser icon next to the property and select CPU -> Usage.

The Alert Trigger Threshold * Alert Monitor Job Interval defines the time interval that condition should hold before generating the alerts. Alert Monitor Job Interval by default is 1 minute but can be adjusted by adding **applicare.alertMonitorJobInterval** (in minutes) property to Applicare.props file. In the example below alert will be generated if the CPU usage is higher than 90 for 3 consecutive polls by Applicare. Similarly in the example below auto clear threshold defines that CPU usage needs to be below 90 for 3 consecutive polls for generating clear alert.

> [Screenshot 2 could not be downloaded during import — see migration report]

Intellicheck Alerts Configuration

IntelliCheck alerts are paused for **intellicheckAlertsPauseTime** (in minutes) after # of alert reaches the **intellicheckAlertPauseThreshold** limit set in Applicare.properties. With the example values below 3 IntelliCheck alerts would come for JDBC resource not available and then alerts on the same JDBC resource would be paused for 30 minutes.

```
intellicheckAlertsPauseTime=30

intellicheckAlertPauseThreshold=3
```

By default, **intellicheckAlertsPauseTime** is set to 15 minutes alert and **intellicheckAlertPauseThreshold** is set to 5 alerts.

Email Notifications

To start the receiving email notifications for new alerts, you first need to configure the SMTP server details.

Navigate to **Applicare** > **Configuration** > **Configure Email/SMTP Configuration** from the top menu.

> [Screenshot 3 could not be downloaded during import — see migration report]

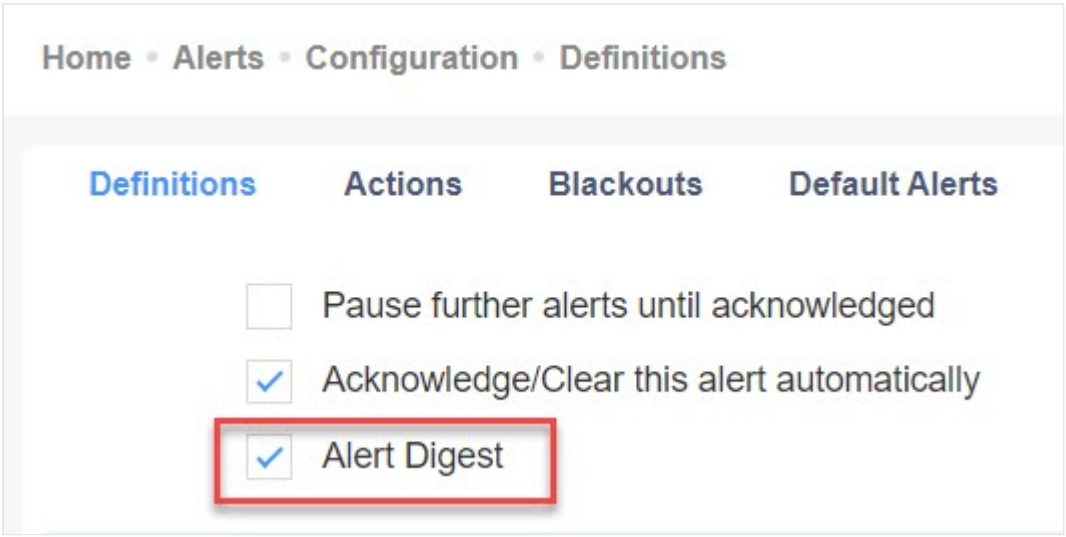
Mail To is the catch all email address. All Applicare alerts are sent to this email address.

To customize who gets what alerts select **Home** > **Alerts** (tab)> **Configuration** (wheel icon). Then select **Actions**.

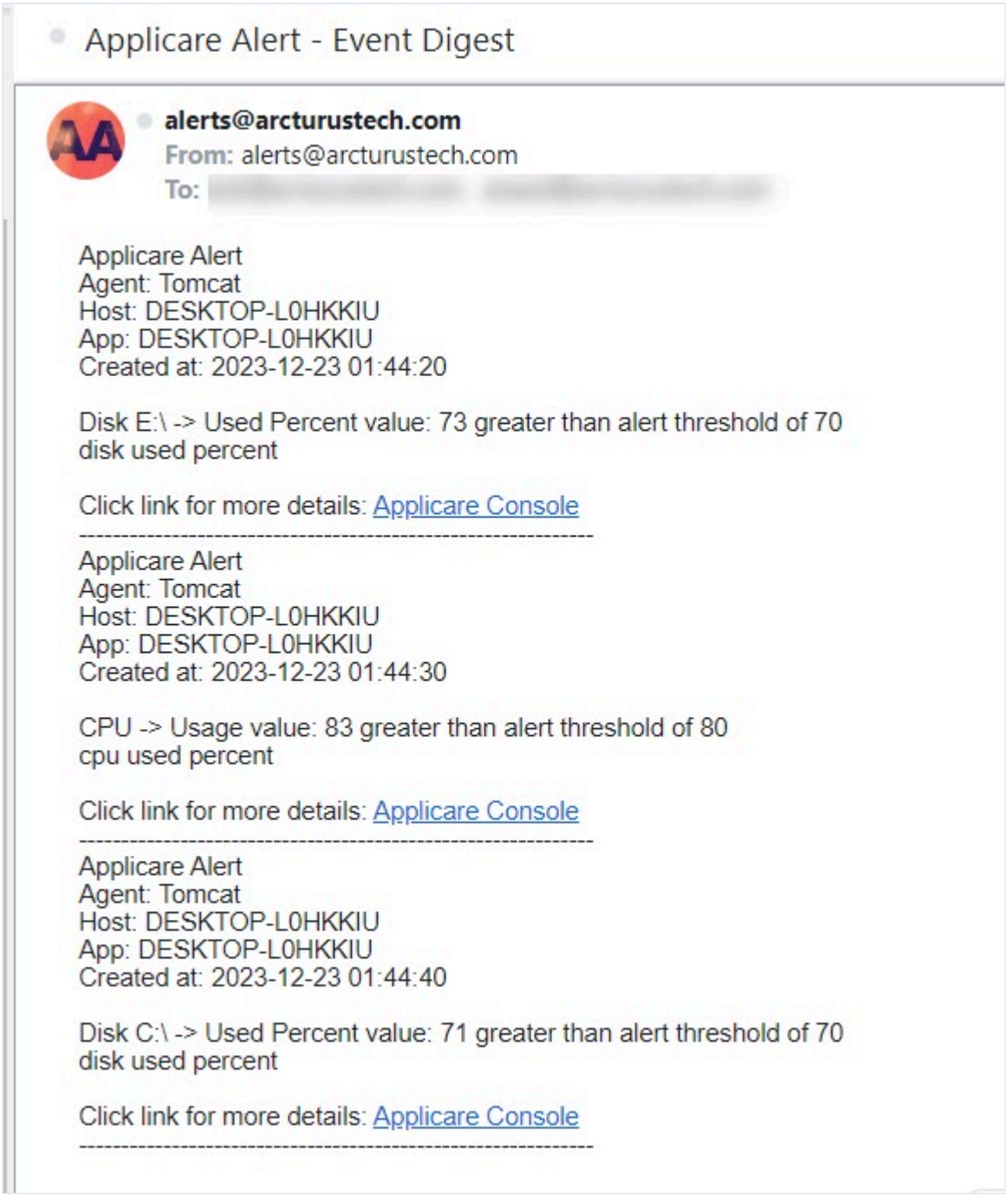
Under actions you can select users, add email addresses or email list addresses for receiving alerts on an application, a server or an individual alert. Applicare would send alert to emails/users defined for individual alert. If there are no specific emails/users defined for individual alerts, it would inherit the same from the server or the application it is part of e.g. If **portalLX** application alerts are sent to Larry and no other specifics are defined for servers that are a part of **portalLX** or individual alerts then all the alert emails for portalLX would be sent to Larry. Once a specific email address/s and users are defined for either server alerts or individual alerts then that overrides the application alert settings.

If you update the emails/users settings are changed for an application after defining setting for individual alerts or server alerts, they would be overwritten by the newly defined settings for the application.

Alert Digest



Applicare includes an alert digest feature that consolidates multiple alerts into a single email. This approach simplifies the management and readability of alerts, providing users with a more organized and user-friendly experience.



Advanced Options

From the configuration screen, you can enable or disable alerting.

Select **Applicare** > **Configuration** > **Advanced Options** from the top menu.

> [Screenshot 6 could not be downloaded during import — see migration report]

Using A Custom Alert Processing Class

You can create a Java class to which will be invoked for each alert created by Applicare agents by implementing the interface **com.arcturus.alerts.notifications.AlertAction** and the only method in the interface public void **processAlert(AlertQueue alert)**. You can get the details of the alert using the getter methods of AlertQueue object as shown in the example class below.

```
TEXT
package com.arcturus.alerts.notifications;

import java.text.SimpleDateFormat;

import com.arcturus.alerts.domain.AlertQueue;

public class SampleAction implements com.arcturus.alerts.notifications.AlertAction{

    public void processAlert(AlertQueue alert) {
        System.out.println("Server: " + alert.getServer()); //the server name where the alert
was generated
        System.out.println("Type: " + alert.getSubsystem()); //type of the alert (JMX_ALERT,
PLATFORM_ALERT, etc..)
        System.out.println("Priority: " + alert.getPriority()); //priority of the alert
        System.out.println("Message: " + alert.getMessage()); //user defined message in alert
definition (description)
        SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        System.out.println("Created: " + dateFormat.format(alert.getCreated())); //time the
alert was created
    }
}
```

To deploy the class and any dependent classes, you can package it in a jar file and copy to **Applicare/server/ext** folder (create it if it doesn't exist) and restart Applicare server.

When compiling make sure to include the **Applicare/server/webapps/applicare/WEB-INF/classes** folder in your classpath.

After restarting the server, navigate to **Applicare** > **Configuration** > **Advanced Options.** Enter the fully qualified class name of your custom class in the "Custom Alert Processing Class:" field and click **Apply**.

> [Screenshot 7 could not be downloaded during import — see migration report]