

SolarWinds Technical Reference

Integrating SolarWinds Orion and Trouble Ticket Systems

Orion Trouble Ticket System Integration	1
Understanding the Requirements	2
Implementing a Test Plan	3
Transitioning to Production	5
Automated Acknowledgement of Alerts	5
Integrating TTS Web Functions into the Orion Web Console	5
About SolarWinds	6

This paper examines the best practices for the automatic opening of trouble tickets from SolarWinds Orion Network Performance Monitor.

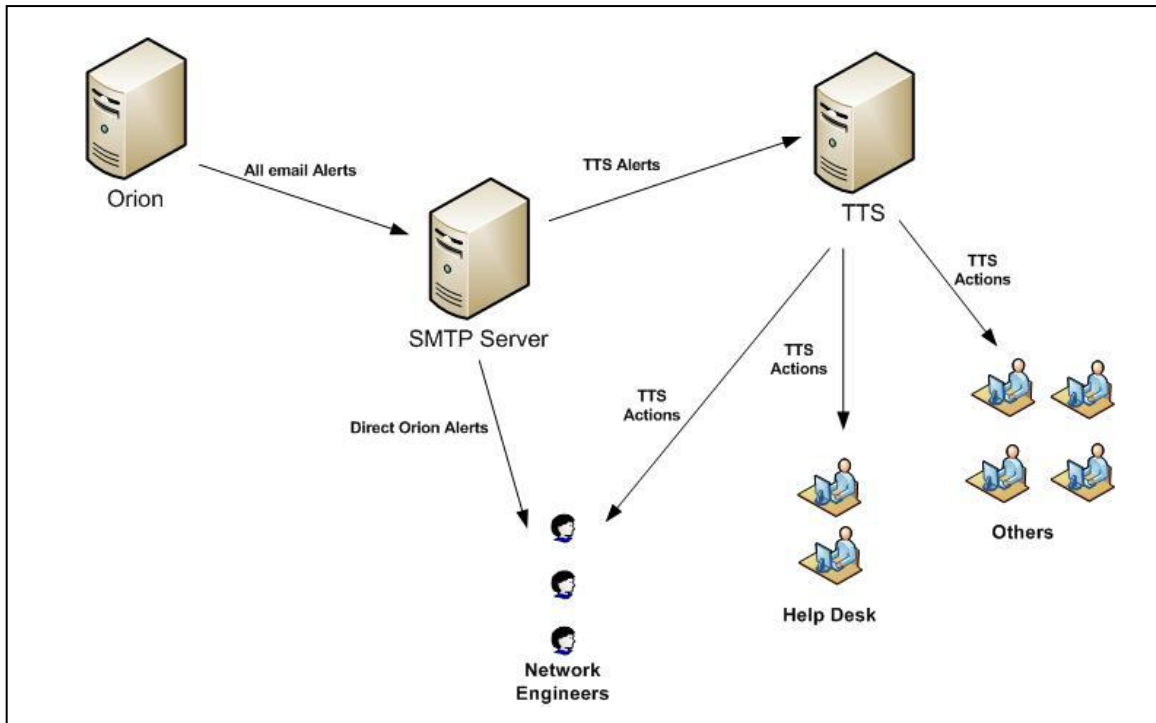
Copyright© 1995-2010 SolarWinds. All rights reserved worldwide. No part of this document may be reproduced by any means nor modified, decompiled, disassembled, published or distributed, in whole or in part, or translated to any electronic medium or other means without the written consent of SolarWinds. All right, title and interest in and to the software and documentation are and shall remain the exclusive property of SolarWinds and its licensors. SolarWinds Orion™, SolarWinds Cirrus™, and SolarWinds Toolset™ are trademarks of SolarWinds and SolarWinds.net® and the SolarWinds logo are registered trademarks of SolarWinds All other trademarks contained in this document and in the Software are the property of their respective owners.

SOLARWINDS DISCLAIMS ALL WARRANTIES, CONDITIONS OR OTHER TERMS, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, ON SOFTWARE AND DOCUMENTATION FURNISHED HEREUNDER INCLUDING WITHOUT LIMITATION THE WARRANTIES OF DESIGN, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL SOLARWINDS, ITS SUPPLIERS OR ITS LICENSORS BE LIABLE FOR ANY DAMAGES, WHETHER ARISING IN TORT, CONTRACT OR ANY OTHER LEGAL THEORY EVEN IF SOLARWINDS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Document Revised: 08/13/2010

Orion Trouble Ticket System Integration

The most common method of integrating Orion with a trouble ticket program is through the use of email messages from Orion to the Trouble Ticket System (TTS). This paper offers guidelines and best practices for using email to open trouble tickets. Below is a diagram of this integration.



Orion Direct Alerts and TTS Action Alerts

The above diagram shows the Orion Network Performance Monitor (NPM) server and any installed modules in the upper left. In this example, alerts on the Orion server have been configured to send email to the SMTP server. The SMTP servers forwards the email to the proper party as specified in the mail to: field. Some of these emails are addressed to the TTS and some are addressed to the network engineers. The TTS has been configured create actions and make assignments based upon input from the alert emails.

The alert notification emails created by Orion are a function of the Orion Advanced Alert Engine. For a more complete explanation of Orion alerts please see [Understanding Orion Advanced Alerts](#). This paper focuses on a practical methodology for implementing automated TTS tickets using email from Orion.

Understanding the Requirements

Trouble Ticket Systems (TTSs) tend to be very complex and powerful, requiring administrators to be experts in the TTS software. TTSs use parsers to extract the trouble ticket information from email and other sources. While the required information is similar from one TTS to another, there are some specific requirements from each vendor, and these may even be TTS version-specific requirements. The alert email contents need to be clearly understood by the TTS system administrator and of the Orion administrator. A lot of repetitive guesswork can be eliminated by coordinating the requirements ahead of time between the two parties. The requirements for Orion and the TTS may include the following:

- Name and network information of the SMTP server to be used for emails.
- “To” address for the TTS to receive emails.
- A possible “Reply to” email address for the TTS (to indicate an alert was received).
- Types of events the Orion system will be alerting to the TTS.
- Capabilities and format of an Orion email alert.
- Event criticality.
- Requirements for the TTS to accept the email text.
- Static information required in the email subject or body for the TTS. This may include severity level, impact,
- Dynamic (variable) information the TTS can expect from Orion such as node id, IP address, contact information and the alert trigger time for the affected device. Variables are denoted using the following format - `${variable}`
- Alert reset actions to clear automated alerts where the alert conditions now longer apply.
- Any requirements to automatically acknowledge alerts.

These requirements may look something like this for a WAN node down TTS alert:

```
Email Subject: ${NodeName} at ${IPAddress}went down at ${AlertTriggerTime}
```

```
Email body:
```

```
Impact: 2
```

```
Urgency: 2
```

```
Priority: 3
```

```
ContactType: monitoring service
```

```
OpenedBy: Solarwinds
```

```
Category: network
```

```
SubCategory: wan
```

```
Type: access failure
```

```
AssignmentGroup: network
```

In the above example, the variables in the email subject will be filled in by Orion for each triggered alert to show the actual node name, IP address, and the time the alert triggered. The email body contains several pieces of static text information specific to this alert. Typically, this example would be configured and implemented as a WAN-only alert. One way to apply alerts to WAN interfaces or WAN routers only is to create a custom property to specify these, such as the yes/no custom property `WAN_Routers`. This can be helpful in defining the static information in the alerts and properly qualifying the alerts to device types and other criteria. The Orion NPM Administrator’s Guide details how to create and apply custom properties.

The static text information shown above is intended for use by the TTS and so would be unnecessary for a direct Orion email alert.

Use of a alert variable to uniquely and unambiguously define the device (such as IP address) is recommended. Device identifiers such as node name can be subject to formatting issues, such as case sensitivity, and may cause device identification errors.

Also note that alerts sent to the TTS can be cleared using an alert reset action. For example, the reset alert action subject could be set to read:

```
Email Subject: ${NodeName} at ${IPAddress} returned to service as ${nodeStatus}
```

To automatically clear alerts in email integration follow these steps:

1. Create the alert trigger and trigger action to communicate the alert to the TTS via email.
2. Apply a reset condition to the alert.
3. Create an email reset action to communicate the new status of the node to the TTS.
4. Apply logic as required in the TTS to clear the ticket or change the ticket status.

In some environments the change of status as indicated by the reset action is enough to close a ticket. In other environments the ticket may be reassigned to be reviewed by an engineer for verification and manual closure.

Implementing a Test Plan

Once all the requirements are approved, the next logical step is to implement test alerting integration. One consideration in implementing TTS integration is the scope of the problem alerting recipients from Orion versus the scope of alerting action recipients from the TTS. Typically these are coordinated to be complementary, so that Orion is directly providing alerts for those issues that are out of scope of TTS requirements. These may include items such as:

- Orion test alerts.
- Alerts for systems that do not impact end users.
- Warning alerts for exceeded thresholds.
- Alerts for parties outside the TTS user domain, such as third party staffing or maintenance.
- Other types of alerts for which having the TTS issue a trouble ticket does not add value.

While it is true that direct Orion alerts and TTS alerts are not required, both do have unique abilities and so both are typically used in production environments. With this in mind, we will create a test plan that takes advantage of these types of alerts as well as some Orion Advanced Alerting features.

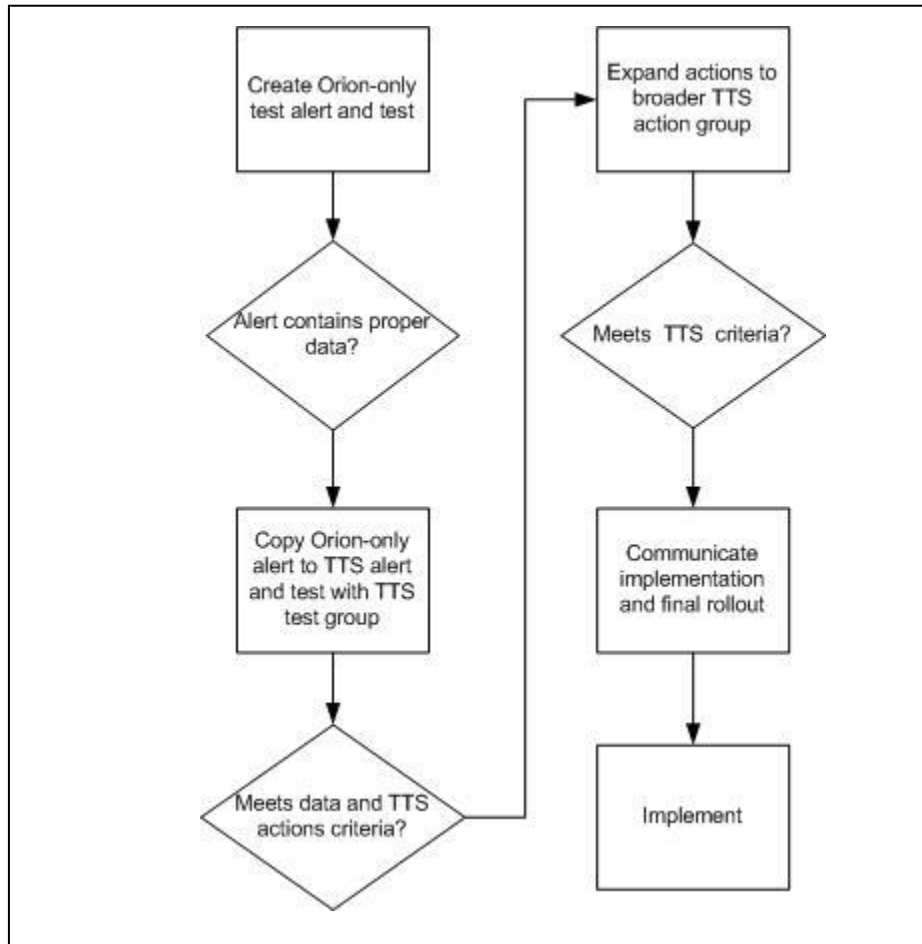
The test plan should include the steps that will be taken to ensure that the Orion/TTS tickets are implemented using a methodology that yields the maximum benefit and causes the least disturbance. Consider what the outcome could be if all the TTS actions were fully enabled and the system began creating Orion tickets with TTS email integration. From the beginning all the parties that would be receiving TTS actions would start receiving them from the TTS. This will only be successful if the alerts and actions are set up perfectly and require no tuning or adjustments. If adjustments are required, all of the recipients of the TTS actions may be impacted by the “churn” of tuning alerts and the associated TTS actions. A better plan of action would be to test the alerts and TTS actions systematically before putting the TTS integration into production.

The final test plan should include the following elements:

- Purpose for initial alert(s) to be created and tested.
- Orion-only initial alert test targets.
- Criteria for success in initial Orion-only alerts.

- Migration of Orion-only initial test alert to TTS test group only.
- Criteria for success of integration TTS test alert and action(s) to test group.
- Expansion of TTS test alert actions to broader parties.
- Criteria for acceptance of broader test.
- Final decision to implement first TTS alert and associated communications.
- Requirements for rollout of subsequent TTS alerts and actions.

The completed test plan flow may look similar to the following:



Sample TTS Implementation Plan

Advanced Alert Manager allows you to copy alerts and rename the copy. This capability may save a lot of time in testing alerts, actions and email destinations. In each step of the TTS alert creation and testing the previous alert can simply be copied and edited to include the new functionality. When doing, care should be taken to rename and enable the new copy and disable the old copies of the alerts that are no longer to be used. You may choose to retain both an Orion-only alert for a small group and the TTS enabled alert for the broader audience. This can provide a backup system to notify critical parties of network outages should TTS problems occur.

Transitioning to Production

Once it has been demonstrated that TTS alerts and actions meet the requirements set out in the planning phase, it is time to implement the alerts. At this point it is not required to go through the initial testing procedure with every alert. It is recommended that testing with a limited TTS audience be required before any alert is rolled out.

Again, email integration is only one of the methods that can be used to integrate Orion with a TTS platform. Email integration is often chosen because it offers easy integration using existing systems. Some Orion customers have created script-based solutions for integrating alerts. While these solutions can add some increased flexibility, they are quite a bit more complex than email integrations. For more information on these integration solutions please see:

<http://thwack.com/forums/p/20551/84948.aspx>

Automated Acknowledgement of Alerts

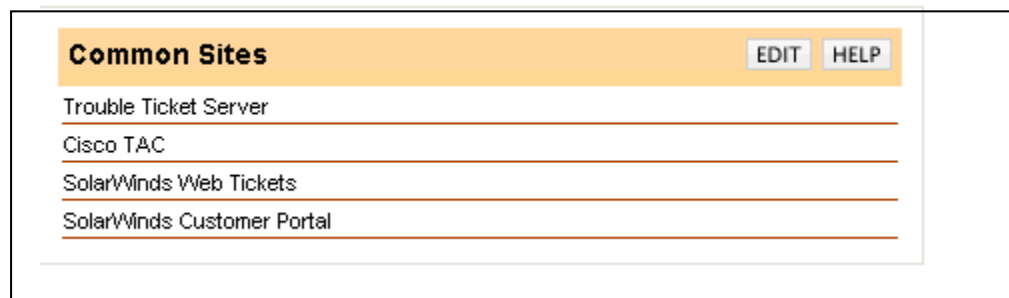
From the Information Flow graphic shown on page 1, one can see that the direction of integrated email information flow is one way, from the Orion server to the SMTP gateway to the TTS. The Orion SDK can be used to automatically set the alert status to acknowledged in Orion once a TTS email has been sent. The SDK is a special developer tool which is not intended for the use of typical Orion users. Use of the SDK requires:

- A request to receive the SDK kit.
- A demonstrated ability to use a Developer's Tool Kit without the need for tutoring or special instruction in a developing tool environment.

Integrating TTS Web Functions into the Orion Web Console

Most TTS platforms offer some sort of web interface. These may be read/write interfaces that allow for creating or updating tickets, or read only, that allow only viewing ticket information. The abilities of these web interfaces varies from vendor to vendor. Orion offers two methods of integrating these with the Orion web console. The first of these is to include the URL for the TTS in a User Links resource added to an Orion View.

In the below sample, a User Link has been added to the Orion NPM Home Page to allow for quick access to the TTS and other common sites. The steps for creating this User Links resource are detailed in the Orion NPM Administrator's Guide, in the Resource Configuration Examples section.



Some TTS platforms also supply a web ticket status interface that may be included in an Orion view as a custom HTML resource. Chances are that this web interface will require authentication, but that should not interfere with the functionality of the resource once the credentials are entered. Some TTS platforms may also allow for a pass-through authentication within the custom HTML. The example below is a mock-up of a possible TTS custom HTML resource.

Current Trouble Tickets
EDIT HELP

READ ONLY

Ticket Management Web

My Requests

6 entries returned - 6 entries matched

Request ID	Summary	Status	Submit Date	Urgency
In Process	Install Hardware	New	17 minutes ago	2-High
IN000000000534	Router 'AUS3500' down	In Progress	20 minutes ago	1-Critical
IN000000000501	VPN config for sales	Completed	2 hours ago	3-Medium
IN000000000499	Interface 'AUS3500 0/6' down	Completed	2 hours ago	1-Critical
IN000000000488	Push IOS for 'Dev6500-Phx	Completed	3 hours ago	3-Medium
IN000000000481	Mow the lawn	Over Due	16 days ago	4-Info Only

For specific information on your TTSs web abilities, refer to the manufacturer's documentation.

About SolarWinds

SolarWinds is rewriting the rules for how companies manage their networks. Guided by a global community of network engineers, SolarWinds develops simple and powerful software for managing networks, small or large. Our company culture is defined by passion for innovation and a philosophy that network management can be simplified for every environment.

SolarWinds products are used by more than one million network engineers to manage IT environments ranging from ten to tens of thousands of network devices. Comprised of fault and performance management products, configuration and compliance products, and tools for engineers, the SolarWinds product family is trusted by organizations around the globe to design, build, maintain, and troubleshoot complex network environments.

SolarWinds is headquartered in Austin, Texas, with sales and product development offices around the world. Join our online community of experts at thwack.com.