

# Application Note: savVi Event Dispatcher



**Contents**

<b>1. Introducing savVi Event Dispatcher.....</b>	<b>3</b>
<b>2. Prerequisites.....</b>	<b>3</b>
<b>3. Installing savVi Event Dispatcher.....</b>	<b>3</b>
<b>4. Enabling savVi Event Dispatcher’s Email Notifications .....</b>	<b>4</b>
<b>5. Customizing the Email Notification.....</b>	<b>6</b>
<b>6. Configuring the Event Dispatcher to Send Events Using Alternative Protocols .....</b>	<b>7</b>
<b>6.1 Example: SMS .....</b>	<b>7</b>
<b>7. Configuring savVi Event Dispatcher to ignore specific sites .....</b>	<b>8</b>
<b>8. Troubleshooting savVi Event Dispatcher .....</b>	<b>9</b>
<b>Appendix: savVi Event Dispatcher Parameters .....</b>	<b>10</b>

## 1. Introducing savVi Event Dispatcher

This Application Note describes the savVi Event Dispatcher feature of Agent Vi's real-time analytics software **savVi**.

savVi Event Dispatcher is an integrated module of savVi, featuring built-in capability to send email notifications for detected analytics events. It can be further customized to dispatch events via *any protocol or API to any destination* such as:

- mobile phones / pagers
- 3<sup>rd</sup> party VMSs

See also [Configuring the Event Dispatcher to Send Events Using Alternative Protocols](#).

## 2. Prerequisites

- savVi version 5.4 or 5.5 must be installed
- Visual Studio, WinDbg or any other debugging tool shouldn't be installed on the PC
- In Windows Programs panel, verify if ActivePython is already installed. In such case – uninstall it.

## 3. Installing savVi Event Dispatcher

### ➤ To install the savVi Event Dispatcher

- After installing savVi, run the installer executable file that you received from Agent Vi; the **Installation Wizard** opens:

Follow the steps of the **Installation Wizard** until installation is complete; the savVi Event Dispatcher files are copied to:

**<savVi Installation Folder>\vi-server\EventDispatcher\**

If you installed savVi in the default location, you will find these files at: **C:\Program Files (x86)\AgentVi\savVi\vi-server\EventDispatcher\**

The installation creates a **Service** named **VIEventDispatcher Service**. It runs by default after installation. To temporarily halt operation of the **savVi Event Dispatcher**, stop the service.

- Update Event Dispatcher configuration (if required) as follows:

In the server on which the event dispatcher was installed, in the **<savVi Installation Folder>\vi-server\EventDispatcher\** folder edit the file **EventDispatcher.conf** file as required based on these conditions:

- a. If the Event Dispatcher is not installed on the server running the primary LBS, change the line "PrimaryLBSAddress"="localhost" to indicate the name or IP of the server running the primary LBS
- b. If a secondary LBS is installed, change the line "SecondaryLBSAddress"="?" to indicate the name or IP of the server running the secondary LBS
- c. If the web service port used by the LBS was changed from the default port number 8080, change the line "WebServicePort"=dword:1f90 to indicate the correct web service port number. Note: the port number value must be a hexadecimal value. For example, if the decimal port number should be set to 2600, the line should be modified to : "WebServicePort"=dword:A28
- d. Save the modified EventDispatcher.conf file
- e. Restart the Event Dispatcher web service for the modified configuration to take effect.

### ➤ Upgrading from an older Event Dispatcher installation

In case you have an older Event Dispatcher installation that worked with Vi-System, please follow these steps:

1. Go to the Event Dispatcher folder under Vi-System installation, typically:  
**C:\Program Files (x86)\AgentVi\Vi-System\vi-server\EventDispatcher\**
2. Backup any configuration files and scripts that you have previously created or edited. More specifically:
  - a. event\_dispatcher.ini
  - b. send\_email.ini – in case you have used the email service
  - c. Python scripts that you created previously
3. Uninstall (Vi-System's) Event Dispatcher
4. Install savVi (migrate from Vi-System to savVi)
5. Install savVi Event Dispatcher
6. Copy over the files that you have backed up to the Event Dispatcher folder, typically: **C:\Program Files (x86)\AgentVi\savVi\vi-server\EventDispatcher\**

## 4. Enabling savVi Event Dispatcher's Email Notifications

savVi Event Dispatcher provides a built-in capability to send email messages whenever savVi triggers an analytics detection event.

The email includes text describing the event details (e.g., camera identification and the analytics rule), as well as the event image showing the camera's field of view at the time of the event.



Email is enabled by default by the line **1=send\_email.py** in the file *event\_dispatcher.ini*.  
[Email messages can be disabled by commenting this line as follows: **;1=send\_email.py**]

➤ **To configure savVi Event Dispatcher email messages**

- Open the installation folder where the savVi Event Dispatcher files are located (see [Installing savVi Event Dispatcher](#))
- Open the file *send\_email.ini* with Notepad; you'll view the following content when you edit the file for the first time:

```
[email]
To=semicolon-separated list of email recipients
From=no-reply@mycompany.com
Subject=savVi Analytics Event
host=
username=
password=
;NOTE: to send via GMail's SMTP server ( smtp.gmail.com ), add
these two optional values: port=587 (instead of the default
port) and TLS=1 (to use TLS)
;port=587
;TLS=1
```

- Define the fields in this file:

Field	Description
<b>To</b>	Specify one or more email recipients to whom to send the events. Example: <a href="mailto:username@gmail.com">username@gmail.com</a> . Separate multiple recipients with a semi-colon.
<b>From</b>	Specify the email sender, e.g., the name of the enterprise secured by the surveillance network (optional); for identification enhancement.
<b>Subject</b>	Specify the email subject, e.g., <b>Analytics Event</b> ; for identification enhancement
<b>host</b>	Specify your email server name (SMTP server name) or IP address. Example: <b>host=</b> smtp.myinternetprovider.com. Obtain it from your system administrator. For Gmail, Hotmail or any other web-based email service, refer to each service's support page.
<b>Username</b>	Specify a valid email user when necessary; it may not be necessary when using your ISP's SMTP server
<b>Password</b>	Specify the user password; it may not be necessary when using your ISP's SMTP server
<b>Port</b>	The SMTP port to be used. If not specified, the default (25) is used.
<b>TLS</b>	Set to 1 if TLS security is to be used. The default is 0 (TLS not used).

After defining these fields, savVi Event Dispatcher begins sending event notifications to your email address.

## 5. Customizing the Email Notification

Users can customize the email notification received when an event is sent from savVi to their mailbox, to suit personal preference.

### ➤ To customize your email notification

1. Open the file `send_email.py` in an editor
2. In the code, go to the procedure `process_events_list`
3. You can modify the variable `text` to include any of the event parameters associated with the Event Dispatcher. See the [Appendix](#) for savVi Event Dispatcher parameters and their descriptions.

## 6. Configuring the Event Dispatcher to Send Events Using Alternative Protocols

**savVi Event Dispatcher** enables sending events using additional communication protocols and generally enables executing external applications. This is achieved by writing a **Python script** that **Event Dispatcher** executes. All event parameters are passed to the **Python script** which is then used to trigger alternative communication protocols (commonly HTTP) or to trigger an external application.

### Note

**Event Dispatcher** will execute all Python scripts listed in the file *event\_dispatcher.ini*.

### 6.1 Example: SMS

The example below demonstrates how **savVi Event Dispatcher** can be configured to send **SMS messages** via a **web-to-SMS gateway service**.

Use the **provided Email script as a baseline** and modify it for sending SMSs.

Alternatively, you can opt to write the **Python script** from scratch.

- **To configure savVi Event Dispatcher to dispatch SMSs via a web-to-SMS gateway service**
- 4. Open the file *send\_email.py* in an editor
- 5. In the code, go to the procedure **process\_events\_list** and change the **process\_events\_list** procedure as follows:

```
for ev in events:
    if ev.has_key('TextMessage'):
        continue
    format = "%s" event received on %s at %s from sensor %s.\n'
    text = format % ( ev['RuleName'],ev['Date'],ev['Time'],ev['SensorID'] )
    post = {} # web form POST params
    post['telephone']='21276544111' # change phone number(s) here
    post['message'] = text
    http_request = urllib2.Request('http://67.243.75.146/cgi-bin/web_sms')
    auth = base64.encodestring('%s:%s' % ('user58390', 'mypassword'))
    http_request.add_header("Authorization", "Basic %s" % auth[:-1])
    http_request.add_data( urllib.urlencode( post ) )
    try:
        urllib2.urlopen(http_request)
    except:
        continue
    urllib2.urlopen(http_request).read()
```

- **To configure savVi Event Dispatcher to dispatch events as SMSs**
- 6. Save the modified script file under another name, for example, *send\_sms.py*
- 7. To send as SMS messages (only), have a single line: **1= send\_sms.py**  
To send as SMS messages *and* as emails, add the line **2=send\_sms.py** in the file *event\_dispatcher.ini*

See the [Appendix](#) for savVi Event Dispatcher parameters and their descriptions.

## 7. Configuring savVi Event Dispatcher to ignore specific sites

**savVi Event Dispatcher** can be configured to filter out events from selected sites. To enable site filtering, the following steps need to be performed:

1. Locate the *EventDispatcher.conf* file under **\Program Files (x86)\Agent Vi\savVi\vi-server\**
2. Edit the file, locate the "IgnoreEventsFromTheseSiteIDs" key, and insert any number into the key (for example - "IgnoreEventsFromTheseSiteIDs"="1"). This is required to log all savVi site IDs, which are logged only when the "IgnoreEventsFromTheseSiteIDs" key in the *EventDispatcher.conf* file is not empty.
3. Restart the **ViEventDispatcher** service
4. Locate the latest ViEventDispatcher log file under **\Program Files (x86)\Agent Vi\savVi\Logs\** folder. All savVi sites will be listed in the log file, along with the corresponding site IDs.
5. Update *EventDispatcher.conf* file with the site IDs that you wish to filter, separated by commas (for example - "IgnoreEventsFromTheseSiteIDs"="1,2,3,4,10,15")
6. Restart the **ViEventDispatcher** service

## 8. Troubleshooting savVi Event Dispatcher

savVi Event Dispatcher logs are available under the following location:

**\Program Files (x86)\Agent V\savVi\Logs\**

File name syntax is *ViEventDispatcher\_yyyy\_mm\_dd\_part\_xx.log*

The log contains service logging as well as Python script run error logs.

## Appendix: savVi Event Dispatcher Parameters

The table below shows the parameters applicable to the Event Dispatcher and their descriptions.

Parameter	Description
<b>Date and Time</b>	Defines the event's date and time, in <b>YYYY-MM-DD</b> and <b>HH:MM:SS</b> formats respectively
<b>SensorID</b>	Defines the <b>sensor number</b> in savVi
<b>SourceStreamInfo</b>	The <b>camera ID defined in the customer's VMS</b> . Corresponds to parameter <b>External ID</b> viewed and configurable in <b>savVi-Manager</b> .
<b>RuleID</b>	<b>Rule number</b> defined in <b>savVi</b>
<b>RuleGUID</b>	Defines the <b>Rule unique identifier</b> . See <b>Web Service Interface Reference Guide</b> for a detailed list of identifiers.
<b>RuleName</b>	User-defined rule name in savVi. E.g.: <b>Suspicious object on platform</b>
<b>ExternalRuleID</b>	<b>Rule identifier</b> defined in the customer's <b>VMS</b> . Corresponds to parameter <b>External Rule ID</b> configurable in <b>savVi-Manager</b> .
<b>Description</b>	Identifies the rule event specifically, e.g., <b>Moving person detected</b>
<b>EventID</b>	Defines a running <b>event identification number</b>
<b>EventTypeld</b>	Enumerated value corresponding to the event parameter <b>Description</b> . See <b>Web Service Interface Reference Guide</b> for a detailed list.
<b>EventImage</b>	Applicable only for detection events. Defines the <b>event image</b> , showing the camera's field of view at the time the detection event is triggered. The attribute is passed as a JPEG image in hexadecimal representation. The attribute was named <b>Image</b> in previous releases of savVi Event Dispatcher.



### Notice

Copyright © 2003-2020 by Agent Video Intelligence Ltd.

Agent Video Intelligence Ltd. holds the copyright to this manual. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means without prior written consent from Agent Video Intelligence Ltd.

### Disclaimer

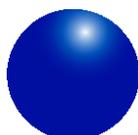
The information in this manual was accurate and reliable at the time of its release for this specific version. However, Agent Video Intelligence Ltd. reserves the right to change the specifications of the product described in this manual without prior notice at any time.

The customer should note that in the field of video there are a number of patents held by various parties. It is the responsibility of the user to assure that a particular implementation does not infringe on those patents. Agent Video Intelligence Ltd. does not indemnify the user from any patent or intellectual property infringement.

### Trademarks

Agent Vi™, Vi™, savVi™, savVi-Analyst™, savVi-Manager™, Vi-Agent™, Vi-AgentProxy™ and Vi-Server™ are trademarks of Agent Video Intelligence Ltd.

All other proprietary names mentioned in this manual are the trademarks of their respective owners.



[www.agentvi.com](http://www.agentvi.com)  
[support@agentvi.com](mailto:support@agentvi.com)