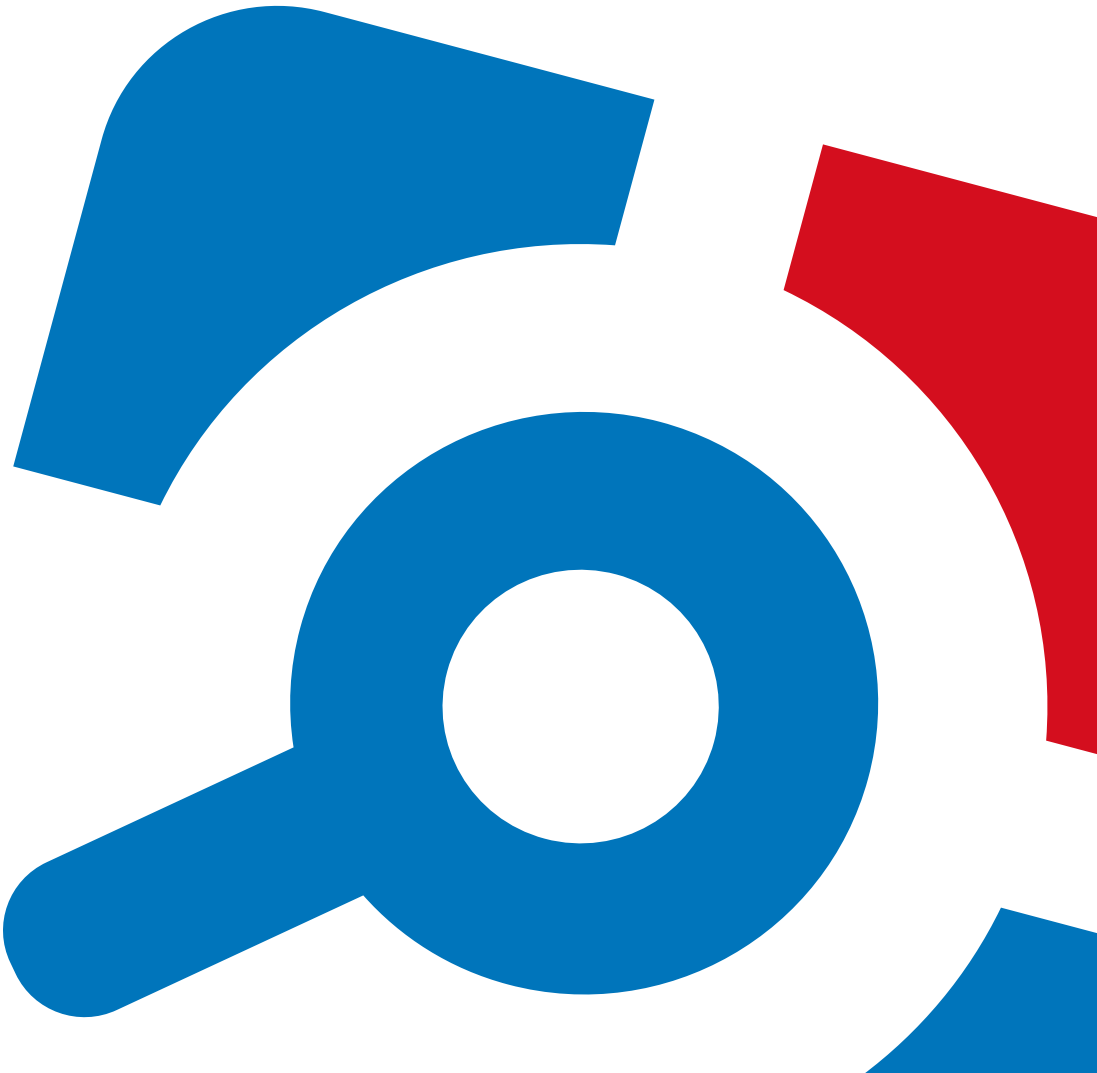
netwrix

Netwrix Auditor Add-on for ServiceNow Incident Management

Quick-Start Guide

Version: 9.6
5/8/2018



Legal Notice

The information in this publication is furnished for information use only, and does not constitute a commitment from Netwrix Corporation of any features or functions, as this publication may describe features or functionality not applicable to the product release or version you are using. Netwrix makes no representations or warranties about the Software beyond what is provided in the License Agreement. Netwrix Corporation assumes no responsibility or liability for the accuracy of the information presented, which is subject to change without notice. If you believe there is an error in this publication, please report it to us in writing.

Netwrix is a registered trademark of Netwrix Corporation. The Netwrix logo and all other Netwrix product or service names and slogans are registered trademarks or trademarks of Netwrix Corporation. Microsoft, Active Directory, Exchange, Exchange Online, Office 365, SharePoint, SQL Server, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks and registered trademarks are property of their respective owners.

Disclaimers

This document may contain information regarding the use and installation of non-Netwrix products. Please note that this information is provided as a courtesy to assist you. While Netwrix tries to ensure that this information accurately reflects the information provided by the supplier, please refer to the materials provided with any non-Netwrix product and contact the supplier for confirmation. Netwrix Corporation assumes no responsibility or liability for incorrect or incomplete information provided about non-Netwrix products.

Table of Contents

1. Introduction

This guide is intended for the first-time users of Netwrix Auditor Integration API add-ons. It can be used for evaluation purposes, therefore, it is recommended to read it sequentially, and follow the instructions in the order they are provided. After reading this guide you will be able to:

- Define add-on parameters
- Execute the add-on
- Review results

NOTE: The add-on works only in combination with Netwrix Auditor so this guide covers a basic procedure for running the add-on and assumes that you have Netwrix Auditor installed and configured in your environment. For installation scenarios, data collection options, as well as detailed information on Integration API, refer to:

- Netwrix Auditor Installation and Configuration Guide
- Netwrix Auditor Administration Guide
- Netwrix Auditor Integration API Overview

1.1. Netwrix Auditor Overview

Netwrix Auditor is a visibility platform for user behavior analysis and risk mitigation that enables control over changes, configurations and access in hybrid IT environments to protect data regardless of its location. The platform provides security analytics to detect anomalies in user behavior and investigate threat patterns before a data breach occurs.

Netwrix Auditor includes applications for Active Directory, Azure AD, Exchange, Office 365, Windows file servers, EMC storage devices, NetApp filer appliances, SharePoint, Oracle Database, SQL Server, VMware, and Windows Server. Empowered with a RESTful API and user activity video recording, the platform delivers visibility and control across all of your on-premises or cloud-based IT systems in a unified way.

Major benefits:

- Detect insider threats—on premises and in the cloud
- Pass compliance audits with less effort and expense
- Increase productivity of IT security and operations teams

2. Netwrix Auditor Add-on for ServiceNow Incident Management Overview

The add-on works in collaboration with Netwrix Auditor, supplying data on suspicious activity or improper actions right to your helpdesk action center. Aggregating data into a single trail simplifies incident processing and handling, makes IT service management more cost effective, and helps address threats as soon as possible.

Implemented as a service, this add-on facilitates the data transition from Netwrix Auditor to ServiceNow ITSM system. The service automatically creates incident tickets in your system and updates them with subsequent events. All you have to do is provide connection details and specify what actions should lead to ticket creation.

On a high level, the add-on works as follows:

1. The add-ons comes with a special set of alerts developed by Netwrix industry experts. With a help of a straight-forward command-line tool, you upload these alerts to Netwrix Auditor and enable integration with add-on.
2. Whenever the alert is triggered, the add-on retrieves an Activity Records for this action using the Netwrix Auditor Integration API. Each Activity Record contains the user account, action, time, and other details.
3. The add-on creates an incident ticket in ServiceNow, populates it with data that was available in the alert, and assigns to a proper team. Now, you can process a ticket as usual.

   To prevent ticket overflow, the service provides an advanced flood suppression mechanism.

For more information on the structure of the Activity Record and the capabilities of the Netwrix Auditor Integration API, refer to Netwrix Auditor Integration API Overview.

3. Use the Add-On

After downloading the add-on package from Netwrix add-on store, copy it to the a computer where Netwrix Auditor Server resides. Unpack the ZIP archive to a folder of your choice; by default, it will be unpacked to the **Netwrix_Auditor_Add-on_for_ITSM** folder.

The main component of the add-on is implemented as a service named **Netwrix Auditor ITSM Integration Service**. This service will run on the computer where Netwrix Auditor Server works, and will use the default Integration API port **9699**. Unless specified, the service will run under the **LocalSystem** account.

To use the add-on, you should check the prerequisites and specify configuration settings, as described in the next sections. After that, run the installer that will apply settings and start the service, as described in the Deploy the Service section.

3.1. Prerequisites

(missing or bad snippet)

On...	Ensure that...
The Netwrix Auditor Server and (missing or bad snippet) side	• The user connecting to Netwrix Auditor Server is granted the **Global administrator** role in Netwrix Auditor or is a member of the **Netwrix Auditor Administrators** group. • The add-on package is copied to a computer where Netwrix Auditor Server resides.
On the ServiceNow side	• ServiceNow version is *Helsinki* or *Istanbul*. **NOTE:** Currently, *Jakarta* version has only experimental support. • A new user is created and has sufficient permissions to create tickets and update them. The **itil** role is recommended. **NOTE:** If you want to reopen closed tickets, you must be granted the right to perform **Write** operations on inactive incidents.

3.2. Define General Add-on Parameters

1. Navigate to your add-on folder and select **ITSMSettings.xml**.
2. Define general parameters such as Netwrix Auditor connection parameters, the number of tickets the service can create per hour, ability to reopen closed tickets, etc. For most parameters, default values are provided. Provide new values as follows: `<parameter>value</parameter>`. You can skip or

define parameters depending on your execution scenario and security policies.

Parameter	Default value	Description
NetwrixAuditorHost	https://localhost:9699	The add-on runs on the computer where Netwrix Auditor Server resides and uses the default Integration API port 9699. To specify a non-default port, provide a new port number (e.g., *https://localhost:8788*). NOTE: The add-on must always run locally, on the computer where Netwrix Auditor Server resides.
NetwrixAuditorUserName	—	Unless specified, the add-on runs under the **LocalSystem** account. If you want the add-on to use another account to connect to Netwrix Auditor Server, specify the account name in the DOMAIN\username format. Alternatively, after deploying the **Netwrix Auditor ITSM Integration Service** service, specify an account in its properties. NOTE: The user running the service and connecting to Netwrix Auditor Server must be granted the **Global administrator** role in Netwrix Auditor or be a member of the **Netwrix Auditor Administrators** group. The user must have sufficient permissions to create files on the computer.
NetwrixAuditorPassword	—	Provide a password for the account. Unless an account is specified, the service runs under the **LocalSystem** account and does not require a password.
TicketFloodLimit	10	Specify the maximum number of standalone tickets the service can

Parameter	Default value	Description
		create during TicketFloodInterval. If a ticket flood limit is reached, the service writes all new alerts into a single ticket.
TicketFloodInterval	3600	Specify the time period, in seconds. During this time period, the service can create as many tickets as specified in **TicketFloodLimit**. The default value is 3600 seconds, i.e., 1 hour.
ConsolidationInterval	900	Specify the time period, in seconds. During this time period, the service does not process similar alerts as they happen but consolidates them before updating open tickets in your ITSM. The default values is 900 seconds, i.e., 15 minutes. This option works in combination with **UpdateTicketOnRepetitiveAlerts** and is helpful if you want to reduce the number of ticket updates on ITSM side. I.e., this option defines the maximum delay for processing alerts and updating existing tickets. Tickets for new alert types are created immediately. For example, a new alert is triggered—the service opens a new incident ticket. The alert keeps firing 20 times more within 10 minutes. Instead of updating the ticket every time, the service consolidates alerts for 15 minutes, and then updates a ticket just ones with all collected data.
CheckAlertQueueInterval	5	Internal parameter. Check and process the alert queue every N seconds; in seconds.
UpdateTicketOnRepetitiveAlerts	true	Instead of creating a new ticket, update an existing active ticket if a similar alert occurs within **UpdateInterval**.

Parameter	Default value	Description
ReopenTicketOnRepetitiveAlerts	true	To open a new ticket for every alert, set the parameter to *"false"*. Instead of creating a new ticket, reopen an existing ticket that is in a closed state (be default, closed, canceled, and resolved) if a similar alert occurs within **UpdateInterval**. This option works only when **UpdateTicketOnRepetitiveAlerts** is set to *"true"*. **NOTE:** If you want to reopen closed tickets, you must be granted the right to perform **Write** operations on inactive incidents.
UpdateInterval	86400	Specify the time period, in seconds. If a similar alert occurs in less than N seconds, it is treated as a part of an existing incident. The default value is 86400 seconds, i.e., 24 hours. If an alerts is triggered after the **UpdateInterval** is over, a new ticket is created.
EnableTicketCorrelation	true	Review history and complement new tickets with information about similar tickets created previously. This information is written to the **Description** field. This option is helpful if you want to see if there is any correlation between past incidents (occurred during last month, by default) and a current incident.
CorrelationInterval	2592000	Specify the time period, in seconds. During this time period, the service treats similar tickets as related and complements a new ticket with data from a previous ticket. The default value is 2592000 seconds, i.e., 1 month. Information on alerts that are older

Parameter	Default value	Description
		than 1 month is removed from internal service storage.
ProcessActivityRecord QueueInterval	5	Internal parameter. Process Activity Record queue every N seconds; in seconds.
DisplayOnlyFirstActivityRecord	true	Add only the first Activity Record in the work notes, Activity Records that update this ticket will be added as attachments to this ticket. If false, all Activity Records will be displayed in the ticket work notes.
ActivityRecordRequestsRetention		
RequestLimit	5000	Internal parameter. The maximum number of Activity Record requests the service can store in its internal memory. Once the limit is reached, the service clears Activity Record requests starting with older ones.
RequestLimitInterval	604800	Internal parameter. The service can store the Activity Record requests not older than N seconds; in seconds. Older Activity Record requests are cleared.
ActivityRecordWebRequests		
RequestLimit	200	Internal parameter. The maximum number of Activity Records the service can retrieve in a single request.
RequestTimeout	180	Internal parameter. By default, 3 minutes. Defines the connection timeout.
TicketRequestsRetention		
RequestLimit	300000	Internal parameter. The maximum number of ticket requests the service can store in its internal memory. Once the limit is reached, the service clears ticket requests starting with older ones.
RequestLimitInterval	604800	Internal parameter. The service can store the ticket requests not older than N seconds; in seconds. Older tickets requests are cleared.

NOTE: Stop and then restart the service every time you update any of configuration files.

3.3. Configure ServiceNow Parameters

1. Navigate to your add-on folder and select **ServiceNowSettings.xml**.
2. Define parameters such as ServiceNow connection parameters inside the `<Connection>` section. New values are provided as follows: `<paramenter>value</parameter>`.

<Connection> parameter	Default value	Description
URL	—	Provide a link to your ServiceNow system(e.g., *https://enterprise.service-now.com*).
UserName	—	Specify a user account. Make sure the user has sufficient permissions to create tickets and update them. The **itil** role is recommended. NOTE: If you want to reopen closed tickets, you must be granted the right to perform **Write** operations on inactive incidents.
Password	—	Provide a password.

3. Review `<TicketParameters>` section. The parameters inside this section correspond to ServiceNow ticket fields and use the same naming (e.g., priority, urgency). To find out a field name in ServiceNow, switch to XML view (on the ticket header, navigate to **Show XML**).

   Each `<TicketParameter>` includes the `<Name></Name>` and `<Value></Value>` pair that defines a ServiceNow ticket field and a value that will be assigned to it. For most parameters, default values are provided. Add more ticket parameters or update values if necessary.

   NOTE: The template remains the same for all alerts and cannot be adjusted per individual alerts.

Name	Value	Description
short_description	[Netwrix Auditor] %AlertName%	Sets **Short description** to alert title (e.g., *[Netwrix Auditor] ITSM Add-On: User Account Locked Out)*.
category	software	Sets the incident **Category** to *"Software"*.
impact	1	Sets **Impact** to *"1 – High"*.
urgency	1	Sets **Urgency** to *"1 – High"*.
severity	1	Sets **Severity** to *"1 – High"*.
assignment_group	d625dccec0a8016700a22a0f7900d06	Sets **Assignment group** to *"Service Desk"*. NOTE: You cannot use a group name as a value. Provide its guid instead.
description	%AlertDescription% %PreviousTicketReference%	Provides an alert description and references to related tickets in **Description**.
work_notes	Alert Details:	Adds the full alert text to **Work notes**, including

Name	Value	Description
	....	data source, who, what, where, etc. To find out what is included in the alert details, see the **ServiceNowSettings.xml** file. NOTE: You can write alert details in the **Additional comments** field instead of **Work notes**. To do this, rename `<Name>work_notes</Name>` into `<Name>comments</Name>`. If you want to write alert details into both fields, create a copy of `<TicketParameter>` entry containing **work_notes** and `<Name>work_notes</Name>` into `<Name>comments</Name>`. To skip alert details, remove entries for **work_notes** or **comments**.

4. Review the `<CorrelationTicketFormat>` section. It shows what information about related tickets will be included in your current ticket. Update the template if necessary.

CorrelationTicketFormat	Description
Previous incident for the same alert type: Number: %number% Opened: %opened_at% Assigned to: %assigned_to% Assignment group: %assignment_group% State: %state%	Each `%parameter%` corresponds to a ServiceNow ticket field. The service will automatically substitute these parameters with values from a related ticket. Rearrange fields or add more if necessary. To find out a field name in ServiceNow, switch to XML view (on the ticket header, navigate to **Show XML**).

5. Review the `<ReopenTicketOptions>` section. It defines the tickets the add-on can reopen automatically.

Name	Description
ClosedTicketStates TicketState	Lists ticket statuses. Only tickets with this status can be reopened. By default, resolved, closed, and canceled tickets can be reopened. To specify a new status, provide its ID in the <TicketState> tag (e.g., 8 for canceled).
NewState	Defines a ticket status once it is reopened. By default, new. To specify another status, provide its ID in the <NewState> tag (e.g., 1 for new).

NOTE: Stop and then restart the service every time you update any of configuration files.

3.4. Integrate Alerts with Add-on

The add-on is shipped with a special set of alerts developed by Netwrix industry experts. These alerts are helpful for handling some routine cases that require service manager's attention, e.g., account lockouts, changes to administrative groups. The alerts have preset filters and can be easily uploaded to Netwrix Auditor, and then integrated with the add-on and your ServiceNow system. These alerts have **ITSM Add-on** prefix in their names.

Alternatively, you can integrate any default Netwrix Auditor alerts or your custom-built alerts with the add-on.

By default, none of the alerts are integrated with add-on. To instruct the add-on to create tickets for alerts, you should enable integration. Netwrix provides a command-line tool for enabling integration with the add-on.

NOTE: Make sure to turn on alerting in Netwrix Auditor. You should manually set the state to *"On"* for all alerts you want to integrate with the add-on.

To integrate alerts with the add-on

1. (missing or bad snippet) start the **Command Prompt** and run the **Netwrix.ITSM.AlertsUploaderTool.exe** tool. The tool is located in the add-on folder. For example:

   ```
   C:\>cd C:\Add-on

   C:\Add-on\Netwrix.ITSM.AlertsUploaderTool.exe
   ```

2. Execute one of the following commands depending on your task.

To...	Execute...
Upload alert set shipped with the add- on to Netwrix Auditor	`Netwrix.ITSM.AlertsUploaderTool.exe /UploadTemplates` Once uploaded, the alerts appear in the **All Alerts** list in Netwrix Auditor, their names start with *"ITSM add-on"*. Make sure to set their state to *"On"* (turn them on) manually.
Review alert list and their integration status	`Netwrix.ITSM.AlertsUploaderTool.exe /List` You will see the full list of Netwrix Auditor alerts, with an enabled or disabled integration status for each alert.
Enable integration	`Netwrix.ITSM.AlertsUploaderTool.exe /Update "<Alert Name>" Enable` where `<Alert Name>` is the name of the alert you want to integrate with the add-on. Provide alert names as they appear in Netwrix Auditor. **NOTE:** You can enable integration with one alert at a time. For example: `Netwrix.ITSM.AlertsUploaderTool.exe /Update "ITSM Add-On: User Account Locked Out" Enable`
Disable integration	`Netwrix.ITSM.AlertsUploaderTool.exe /Update "<Alert Name>" Disable`

To...	Execute...
	where `<Alert Name>` is the name of the alert for which you want to disable integration. NOTE: You can disable integration with one alert at a time. For example: `Netwrix.ITSM.AlertsUploaderTool.exe /Update "ITSM Add-On: User Account Locked Out" Disable`

3.5. Deploy the Service

1. Locate the add-on folder on the computer where Netwrix Auditor Server resides.
2. Run the **install.cmd** file. The file deploys and enables the **Netwrix Auditor ITSM Integration Service**.

NOTE: Stop and then restart the service every time you update any of configuration files.

3.6. Configure Integration Service to Use Proxy

If you are using a proxy to provide access to the Internet, consider that **Netwrix Auditor ITSM Integration Service** will need some additional configuration for proxy server to be detected properly. The reason is that this service runs under the **LocalSystem** account (non-interactive), which requires proxy settings to be specified manually, as recommended by Microsoft.

To configure integration service settings

1. Navigate to the add-on folder (default name is **Netwrix_Auditor_Add-on_for_ITSM**) and select **Netwrix.ITSM.IntegrationService.exe.config** service configuration file.

   NOTE: If **Netwrix Auditor ITSM Integration Service** is running, stop it before modifying configuration file.

2. Open this XML file for edit and add the following section:

```
<system.net>
  <defaultProxy>
    <proxy
      proxyaddress="http://<ip_address>:<port>"
      usesystemdefault="True"
      autoDetect="False" />
  </defaultProxy>
</system.net>
```

   Here:

Parameter	Description
proxyaddress	Specify default proxy address and connection port, e.g., *http://172.28.13.79:8080*
usesystemdefault	Set to *True* to allow Internet Explorer proxy settings to be overwritten with custom settings.
autoDetect	Set to *False*.

3. Start **Netwrix Auditor ITSM Integration Service**.

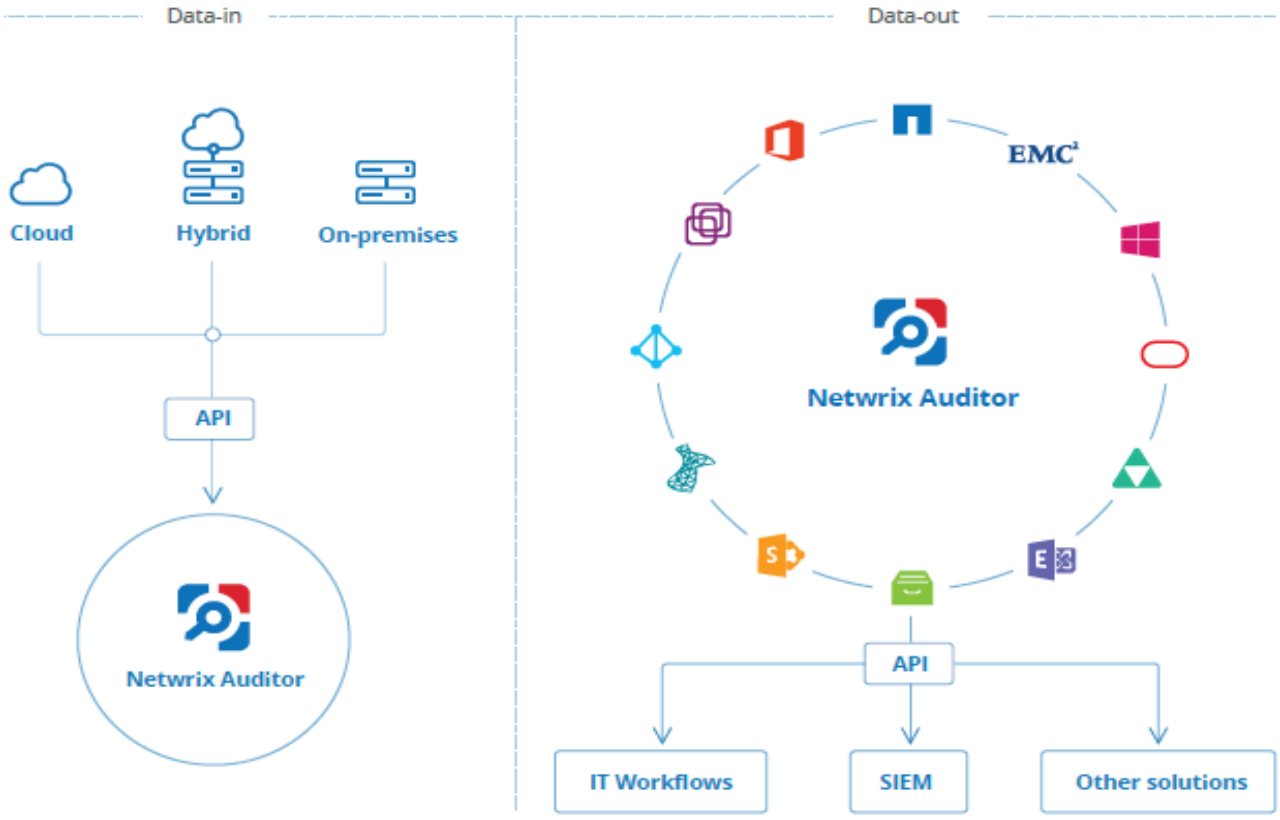
4. Netwrix Auditor Integration API Overview

Netwrix Auditor Add-on for ServiceNow Incident Management leverages Netwrix Auditor Integration API. Although you can always use the add-on as is, but Netwrix encourages customers to create their own integration add-ons. The add-ons created based on Netwrix Auditor Integration API capabilities are easily tailored to your specific environment and business requirements.

Netwrix Auditor Integration API—endless integration, auditing and reporting capabilities.

The Netwrix Auditor Integration API provides access to audit data collected by Netwrix Auditor through REST API endpoints. According to the RESTful model, each operation is associated with a URL. Integration API provides the following capabilities:

- **Data in**: Solidify security and meet regulatory compliance standards by enabling visibility into what is going on in any third-party application.
- **Data out**: Further automate your business processes, IT security and operations workflows by enriching third-party solutions with actionable audit data.



Netwrix Auditor Integration API operates with XML- and JSON-formatted Activity Records—minimal chunks of audit data containing information on *who* changed *what*, *when* and *where* this change was made. XML format is set as default.

With Integration API you can write Activity Records to the SQL Server-based Audit Database and access audit data from remote computers. Also, Netwrix prepares add-ons—sample scripts—to help you integrate your SIEM solutions with Netwrix Auditor.

Netwrix does not limit you with applications that can be used with Integration API. You can write RESTful requests using any tool or application you prefer—cURL, Telerik Fiddler, various Google Chrome or Mozilla FireFox plug-ins, etc.

See Netwrix Auditor Integration API Guide for more information.