Splunk® Enterprise Getting Data In 7.3.2

Monitor data through Windows Management Instrumentation (WMI)

Generated: 10/10/2019 10:43 pm
Monitor data through Windows Management Instrumentation (WMI)

Splunk Enterprise supports the use of Windows Management Instrumentation (WMI) providers for agentless access to Windows performance and event log data on remote machines. You can pull event logs from all the Windows machines in your environment without installing anything on those machines.

If possible, use a universal forwarder rather than WMI to collect data from remote machines. The resource load of WMI can exceed that of a Splunk universal forwarder in many cases. Use a forwarder if you collect multiple event logs or performance counters from each host, or from very busy hosts like domain controllers. See Considerations for deciding how to monitor remote Windows data in this manual. If you have Splunk Cloud, you must use the universal forwarder to collect data from WMI providers and forward it to your Splunk Cloud deployment.

WMI-based data inputs can connect to multiple WMI providers. The input runs as a separate process called splunk-wmi.exe. It is a scripted input.

What do you need to monitor WMI-based data?

Here are the basic minimum requirements to monitor WMI-based data. You might need additional permissions based on the logs or performance counters you want to monitor.

For additional details on what's required to monitor WMI-based data, see "Security and remote access considerations" later in this topic.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Required permissions</th>
</tr>
</thead>
</table>
| Monitor remote event logs over WMI           | * Splunk Enterprise must run on Windows  
* Splunk Enterprise must run as a domain user with at least read access to WMI  
* Splunk Enterprise must run as a domain user with appropriate access to the desired event logs |
| Monitor remote performance monitor counters over WMI | * Splunk Enterprise must run on Windows  
* Splunk Enterprise must run as a domain user with at least read access to WMI  
* Splunk Enterprise must run as a domain user with appropriate access to the Performance Data Helper libraries |
<table>
<thead>
<tr>
<th>Activity</th>
<th>Required permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Security and remote access considerations</strong></td>
<td></td>
</tr>
</tbody>
</table>

Splunk Enterprise and your Windows network must be correctly configured for WMI data access. Review the following prerequisites before attempting to use Splunk Enterprise to get WMI data.

Before Splunk Enterprise can get WMI-based data:

- It must be installed with a user that has permissions to perform remote network connections.
- The user Splunk Enterprise runs as must be a member of an Active Directory (AD) domain or forest and must have appropriate privileges to query WMI providers.
- The Splunk user must also be a member of the local Administrators group on the computer that runs Splunk Enterprise.
- The computer that runs Splunk Enterprise must be able to connect to the remote machine and must have permissions to get the desired data from the remote machine once it has connected.
- Both the Splunk Enterprise instance and the target machines must be part of the same AD domain or forest.

The user that Splunk Enterprise runs as does not need to be a member of the Domain Admins group (and for security reasons, should not be). However, you must have domain administrator privileges to configure access for the user. If you don't have domain administrator access, find someone who can either configure Splunk user access or give domain administrator rights to you.

If you install Splunk Enterprise as the Local System user, remote authentication over WMI does not work. The Local System user has no access to other machines on the network. It is not possible to grant privileges to a Local System account for access to another host.

You can give the Splunk user access to WMI providers by doing one of the following:

- Adding it to the local Administrators group on each member host you want to poll (not recommended for security reasons).
- Adding it to the Domain Admins global group (not recommended for security reasons).
- Assigning least-permissive rights as detailed below (recommended).
**Group memberships and resource access control lists (ACLs)**

To maintain security integrity, place Splunk users into a domain global group and assign permissions on Windows machines and resource ACLs to that group, instead of assigning permissions directly to the user. Assignment of permissions directly to users is a security risk, and can cause problems during security audits or future changes.

**Configure WMI for least permissive access**

If the user you configured Splunk Enterprise to run as is not a domain administrator, you must configure WMI to provide access to this user. Grant only least-permissive access to all Windows resources, including WMI. In order to grant this type of access, follow this checklist. For additional information and step-by-step instructions, see Prepare your Windows network for a Splunk Enterprise installation in the *Installation* manual.

You must grant several levels of access to the user Splunk Enterprise runs as for Splunk Enterprise to collect data over WMI using the least-permissive method:

To deploy these user rights assignments domain-wide, use the Domain Security Policy (dompol.msc) Microsoft Management Console (MMC) snap-in. After deployment, member hosts inherit those rights assignments on the network during the next AD replication cycle. Restart Splunk Enterprise instances on those machines for the changes to take effect.

To extend this access to domain controllers specifically, assign the rights using the Domain Controller Security Policy (dcpol.msc) snap-in.

- **Local Security Policy Permissions.** The Splunk user needs the following Local Security Policy user rights assignments defined on each machine you poll for WMI-based data:
  - Access this Computer from the Network
  - Act as part of the operating system
  - Log on as a batch job
  - Log on as a service
  - Profile System Performance
  - Replace a process level token

- **Distributed Component Object Model (DCOM) configuration and permissions.** DCOM must be enabled on every machine you want to monitor. In addition, the Splunk Enterprise user must be assigned permissions to access DCOM. There are many methods available to do
this, but the best is to nest the "Distributed COM Users" domain global group into the "Distributed COM Users" local group on each machine you want to monitor, then add the Splunk Enterprise user to the "Distributed COM Users" domain global group. See "Securing a Remote WMI Connection" (http://msdn.microsoft.com/en-us/library/aa393266(VS.85).aspx) on MSDN for advanced options to give the Splunk Enterprise user access to DCOM.

**Performance Monitor configuration and permissions.** The Splunk Enterprise user must be a member of the "Performance Log Users" local group in order for Splunk Enterprise to access remote performance objects over WMI. The best way to do this is to nest the "Performance Log Users" domain global group into the "Performance Log Users" local group on each member host and then assign the user to the global group.

**WMI namespace security.** The WMI namespace that Splunk Enterprise accesses (most commonly Root\CIMV2) must have proper permissions. These permissions must be set manually on each host in your enterprise, as there is no global WMI security. Use the WMI Security MMC snap-in (wmimgmt.msc) to enable the following permissions on the WMI tree for each host at the Root namespace for the Splunk user:
- Execute Methods
- Enable Account
- Remote Enable
- Read Security

These rights must be assigned to the Root namespace and all subnamespaces below it. See "Managing WMI security" (https://technet.microsoft.com/en-us/library/cc731011.aspx) on Microsoft TechNet.

**Note:** There is no standard facility for deploying WMI security settings remotely to multiple machines at once using Group Policy. However, Set WMI namespace security via GPO (http://blogs.msdn.com/spatdsg/archive/2007/11/21/set-wmi-namespace-security-via-gpo-script.aspx) on MSDN Blogs offers instructions on how to create a startup script that you can place inside a Group Policy Object (GPO), which sets the namespace security once the GPO applies to the desired hosts. You can then deploy this GPO domain-wide or to one or more Organizational Units (OUs).

**Firewall configuration.** If you have a firewall enabled, you must configure it to allow access for WMI. If you use the Windows Firewall included with
recent versions of Windows, the exceptions list explicitly includes WMI. You must set this exception for both the originating and the target machines. See Connecting to WMI Remotely Starting with Vista (http://msdn.microsoft.com/en-us/library/aa822854(VS.85).aspx) on MSDN for more details.


**Test access to WMI providers**

After you configure WMI and set up the Splunk user for access to your domain, test access to the remote machine.

This procedure includes steps to temporarily change the Splunk Enterprise data store directory (the location SPLUNK_DB points to). You must do this before testing access to WMI. Failure to do so can result in missing WMI events. This is because the splunk-wmi.exe process updates the WMI checkpoint file every time it runs.

If you attempt to log into a domain controller, you might have to change your domain controller security policy to assign the "Allow log on locally" policy for the designated user.

1. Log into the machine Splunk Enterprise runs on as the Splunk user.

2. Open a command prompt (click **Start -> Run** and type **cmd**).

3. Go to the `bin` subdirectory under your Splunk Enterprise installation (for example, `cd c:\Program Files\Splunk\bin`).

4. Determine where Splunk Enterprise currently stores its data by running:

```
> splunk show datastore-dir
```

**Note:** Remember where Splunk Enterprise stores its data. You will recall it later.

5. Run the following command to change where Splunk Enterprise stores its data temporarily:
> splunk set datastore-dir %TEMP%

**Note:** This example sets the data store directory to the current directory specified in the TEMP environment variable. If you want to set it to a different directory, you can do so, but the directory must already exist.

6. Restart Splunk Enterprise:

> splunk restart

**Note:** It might take a while for Splunk Enterprise to restart.

7. Once Splunk Enterprise has restarted, test access to WMI providers, replacing `<host>` with the name of the remote host:

> splunk cmd splunk-wmi -wql "select * from win32_service" -namespace \\
\ `<host>`\root\cimv2

- If you see data streaming back and no error messages, then Splunk Enterprise was able to connect to the WMI provider and query successfully.
- If there is an error, a message with a reason on what caused the error appears. Look for the `error="<msg>"` string in the output for clues on how to correct the problem.

After testing WMI access, point Splunk Enterprise back to the correct database directory by running the following command, and then restarting Splunk Enterprise:

> splunk set datastore-dir <directory shown from Step 4>

**Configure WMI-based inputs**

All remote data collection in Splunk Enterprise on Windows happens through either WMI providers or a forwarder. See Considerations for deciding how to monitor remote Windows data in this manual.

You can configure WMI-based inputs either in Splunk Web or by editing configuration files. More options are available when using configuration files.

**Configure WMI-based inputs with Splunk Web**

To add WMI-based inputs, use the "Remote event log monitoring" and "Remote Performance monitoring" data inputs. See Configure remote Windows performance monitoring with Splunk Web. See also Configure remote Windows event log monitoring
Configure WMI-based inputs with configuration files

wmi.conf handles remote data collection configurations. Review this file to see the default values for WMI-based inputs. If you want to make changes to the default values, edit a copy of wmi.conf in %SPLUNK_HOME%/etc/system/local. Set values only for the attributes you want to change for a given type of data input. See About configuration files in the Admin manual.

wmi.conf contains several stanzas:

- The [settings] stanza, which specifies global WMI parameters.
- One or more input-specific stanzas, which define how to connect to WMI providers to get data from the remote machine.

Global settings

The [settings] stanza specifies global WMI parameters. The entire stanza and every parameter within it are optional. If the stanza is not present, Splunk Enterprise assumes system defaults.

When Splunk Enterprise cannot connect to a defined WMI provider, it generates an error in splunkd.log:

05-12-2011 02:39:40.632 -0700 ERROR ExecProcessor - message from ")C:\Program Files\Splunk\bin\splunk-wmi.exe" WMI - Unable to connect to WMI namespace ")\w2k3m1\root\cimv2" (attempt to connect took 42.06 seconds) (error="The RPC server is unavailable." HRESULT=800706BA)

The following attributes control how Splunk Enterprise reconnects to a given WMI provider when an error occurs.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial_backoff</td>
<td>How long, in seconds, to wait the first time after an error occurs before trying to reconnect to the WMI provider. If connection errors continue to occur, Splunk Enterprise doubles the wait time until it reaches the value specified in max_backoff.</td>
<td>5</td>
</tr>
<tr>
<td>max_backoff</td>
<td>How long, in seconds, to wait between connection attempts, before invoking</td>
<td>20</td>
</tr>
</tbody>
</table>
max_retries_at_max_backoff

If the wait time between connection attempts reaches \texttt{max_backoff}, how many times to try to reconnect to the provider, every \texttt{max_backoff} seconds. If Splunk Enterprise continues to encounter errors, it gives up, and won't attempt to connect to the problem provider again until you restart. It will continue to log errors such as the example shown above.

\begin{table}[h]
\centering
\begin{tabular}{|l|p{12cm}|c|}
\hline
\textbf{Attribute} & \textbf{Description} & \textbf{Default value} \\
\hline
max_retries_at_max_backoff & If the wait time between connection attempts reaches \texttt{max_backoff}, how many times to try to reconnect to the provider, every \texttt{max_backoff} seconds. If Splunk Enterprise continues to encounter errors, it gives up, and won't attempt to connect to the problem provider again until you restart. It will continue to log errors such as the example shown above. & 2 \\
\hline
checkpoint_sync_interval & How long, in seconds, to wait for state data (event log checkpoint) to be written to disk. & 2 \\
\hline
\end{tabular}
\end{table}

\textbf{Input-specific settings}

Input-specific stanzas tell Splunk Enterprise how to connect to WMI providers. They are defined by one of two attributes that specify the type of data Splunk Enterprise should gather. The stanza name can be anything, but usually begins with \texttt{WMI:}, for example:

\texttt{[WMI:AppAndSys]}

When you configure WMI-based inputs in Splunk Web, Splunk Enterprise uses this naming convention for input-specific stanza headers.

You can specify one of two types of data inputs in an input-specific stanza:

- \textbf{Event log}. The \texttt{event_log_file} attribute tells Splunk Enterprise to expect event log data from the sources defined in the stanza.

- \textbf{Windows Query Language (WQL)}. The \texttt{wql} attribute tells Splunk Enterprise to expect data from a WMI provider. You must also specify a valid WQL statement. You must use this attribute when you collect performance data.

Do not define both of these attributes in one stanza. Use only one or the other. Otherwise, the input defined by the stanza will not run.

The common attributes for both types of inputs are:
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>server</td>
<td>A comma-separated list of hosts from which to get data. If this attribute is missing, Splunk Enterprise assumes that you want to connect to the local machine.</td>
<td>The local host</td>
</tr>
<tr>
<td>interval</td>
<td>Tells Splunk Enterprise how often, in seconds, to poll for new data. If this attribute is not present and defined, the input that the stanza defines will not run.</td>
<td>N / A</td>
</tr>
<tr>
<td>disabled</td>
<td>Tells Splunk Enterprise whether this input is enabled or disabled. Set this attribute to 1 to disable the input, and 0 to enable it.</td>
<td>0 (enabled)</td>
</tr>
</tbody>
</table>

The event log-specific parameters are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>event_log_file</td>
<td>A comma-separated list of event log channels to monitor.</td>
<td>N / A</td>
</tr>
<tr>
<td>current_only</td>
<td>Whether or not to collect events that occur only when it is running. If events are generated when Splunk Enterprise is stopped, it will not attempt to index those events when it is started again. Set to 1 to collect events that occur only when it is running, and 0 to collect all events.</td>
<td>0 (gather all events)</td>
</tr>
<tr>
<td>disable_hostname_normalization</td>
<td>Do not normalize the host name that is retrieved from a WMI event. By default, Splunk Enterprise normalizes host names by producing a single name for the host by identifying various equivalent host names for the local system. Set this parameter to 1 to disable host name normalization in events, and 0 to normalize host names in</td>
<td>0 (normalize host names for WMI events)</td>
</tr>
</tbody>
</table>
The WQL-specific parameters are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>wql</td>
<td>A valid WQL statement.</td>
<td>N / A</td>
</tr>
<tr>
<td>namespace</td>
<td>(Optional) Specifies the path to the WMI provider. The local machine must be able to connect to the remote machine using delegated authentication. If you do not specify a path to a remote machine, Splunk Enterprise connects to the default local namespace (\Root\CIMV2). This default namespace is where most of the providers you are likely to query reside. Microsoft provides a list of namespaces for Windows XP and later versions of Windows (<a href="http://msdn.microsoft.com/en-us/library/aa394084(VS.85).aspx">http://msdn.microsoft.com/en-us/library/aa394084(VS.85).aspx</a>).</td>
<td></td>
</tr>
<tr>
<td>current_only</td>
<td>Whether or not an event notification query is expected. See &quot;WQL query types: event notification versus standard&quot; in this topic for additional information. Set this attribute to 1 to tell Splunk Enterprise to expect an event notification query, and 0 to expect a standard query.</td>
<td>0 (expect a standard query)</td>
</tr>
</tbody>
</table>

**WQL query types: event notification versus standard**

The current_only attribute in WQL stanzas determines the type of query the stanza expects to use to collect WMI-based data. When you set the attribute to 1, the stanza expects event notification data. Event notification data is data that alerts you of an incoming event. To get event notification data, you must use an event notification query.

For example, to find out when a remote host spawns processes, you must use an event notification query. Standard queries have no facilities for notifying you when an event has occurred, and can only return results on information that already exists.

Conversely, if you want to know what already-running processes on your system begin with the word "splunk", you must use a standard query. Event notification queries cannot tell you about static, preexisting information.

Event notification queries require that the WQL statement defined for the stanza be structurally and syntactically correct. Improperly formatted WQL will cause the input defined by the stanza to not run. Review the wmi.conf configuration file...
reference for specific details and examples.

**WQL query stanzas do not update the WMI checkpoint file**

When you use a WQL query stanza to gather data through WMI, Splunk Enterprise does not update the WMI checkpoint file - the file that determines if WMI data has been indexed. This is by design - a WQL query of any type returns dynamic data and a context for saving a checkpoint for the data produced cannot be built. This means that Splunk Enterprise indexes WMI data that it collects through WQL query stanzas as fresh data each time the stanza runs. This can result in the indexing of duplicate events and possibly impact license volume.

If you need to index data regularly, such as event logs, use the appropriate monitor on a universal forwarder. If you must use WMI, use a standard WMI query type.

**Examples of wmi.conf**

The following is an example of a *wmi.conf* file:

```
[settings]
initial_backoff = 5
max_backoff = 20
max_retries_at_max_backoff = 2
checkpoint_sync_interval = 2

[WMI:AppAndSys]
server = foo, bar
interval = 10
event_log_file = Application, System, Directory Service
disabled = 0

[WMI:LocalSplunkWmiProcess]
interval = 5
wql = select * from Win32_PerfFormattedData_PerfProc_Process where Name = "splunk-wmi"
disabled = 0

# Listen from three event log channels, capturing log events that occur only
# while Splunk Enterprise runs. Gather data from three machines.
[WMI:TailApplicationLogs]
interval = 10
event_log_file = Application, Security, System
server = srv1, srv2, srv3
disabled = 0
current_only = 1
```
# Listen for process-creation events on a remote machine

[WMI:ProcessCreation]
interval = 1
server = remote-machine
wql = select * from __InstanceCreationEvent within 1 where
TargetInstance isa 'Win32_Process'
disabled = 0
current_only = 1

# Receive events whenever someone plugs/unplugs a USB device to/from the computer

[WMI:USBChanges]
interval = 1
wql = select * from __InstanceOperationEvent within 1 where
TargetInstance ISA 'Win32_PnPEntity' and
TargetInstance.Description='USB Mass Storage Device'
disabled = 0
current_only = 1

## Fields for WMI data

When Splunk Enterprise indexes data from WMI-based inputs, it sets the originating host from the data received. it sets the **source** for received events to **wmi**. It sets the **source type** of the incoming events based on the following conditions:

- For event log data, Splunk Enterprise sets the source type to `WinEventLog:<name of log file>`. For example, `WinEventLog:Application`.
- For WQL data, Splunk Enterprise sets the source type to the name of the stanza that defines the input. For example, for a stanza named `[WMI:LocalSplunkdProcess]`, Splunk sets the source type to `WMI:LocalSplunkdProcess`.

### WMI and event transformations

WMI events are not available for transformation at index time. You cannot modify or extract WMI events as Splunk Enterprise indexes them. This is because WMI events arrive as a single source (a scripted input), which means they can be matched only as a single source.

You can modify and extract WMI events at search time. You can also address WMI-based inputs at parse time by specifying the sourcetype `[wmi]`.

For information on how to transform events as they arrive in Splunk Enterprise, see About indexed field extraction in this manual.
Troubleshooting WMI inputs

If you encounter problems receiving events through WMI providers or are not getting the results you expect, see Common Issues with Splunk and WMI in the Troubleshooting Manual.