

5nine Hyper-V Cloud vMonitor API 1.1

Contents

5nine Hyper-V Cloud vMonitor PowerShell API.....	3
Authentication	3
Logon.....	3
Logout.....	3
Set master credentials.....	3
Monitoring objects.....	3
Add datacenter.....	3
Add cluster	4
Add host	4
Add port	4
Add service	5
Set datacenter.....	5
Set cluster.....	5
Set host.....	6
Set port.....	6
Set service	6
Remove object	7
Set object monitoring status.....	7
Get monitored objects	7
Options commands	7
Get global options.....	7
Set global options.....	7
Get indicators list	8
Get alert definitions	8
Modify alert definition	9
Set credentials.....	9
Results commands	10
Get summary status	10
Get summary results	10
Get triggered alerts	10

Remove triggered alert	10
Get workload data.....	10
Misc commands	12
Get monitoring status	12
Start monitoring.....	12
Stop monitoring	12
Examples	12
REST API.....	14
Authentication	14
LogOn	14
LogOff.....	15
Set master credentials.....	15
Monitoring objects.....	16
Add/Update objects	16
Set objects.....	16
Remove object	17
Set object monitoring status.....	17
Get monitored objects	17
Options commands	17
Get global options.....	17
Set global options.....	17
Get alert definitions	18
Modify alert definition	18
Set credentials.....	19
Results commands	19
Get summary status	19
Get summary results	19
Get triggered alerts	19
Remove triggered alert	20
Get workload data.....	20
Misc commands	20
Get monitoring status	20
Start monitoring.....	20
Stop monitoring	21
Examples	21



OData Feed.....	22
Entities.....	22
List of monitored objects	22
List of triggered alerts	23
List of alert definitions	23

5nine Hyper-V Cloud vMonitor PowerShell API

To add vMonitor snapin to console, use command:

```
add-pssnapin vMonitorAPI
```

Authentication

Logon

```
Enter-Session -Login <String> [-Password <String>]
```

Authenticate to vMonitor API.

Login – (mandatory) the username.

Password – (optional) the password.

Logout

```
Exit-Session
```

Closes current authenticated session.

Set master credentials

```
Set-MasterCredentials [-Login <String>] [-Password <String>]
```

Login – (optional) the username of the predefined master user (by default login is 'admin').

Password – (optional) the password for the master user.

Monitoring objects

Add datacenter

```
Add-DataCenter -Name <String> [-Login <String>] [-Password <String>]
```

Add cluster to monitoring list.



Name – (mandatory) the name of the object that will be displayed in monitoring application.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Add cluster

Add-Cluster [-Name <String>] -Address <String> [-Login <String>] [-Password <String>]

Add cluster to monitoring list.

Name – (optional) the name of the object that will be displayed in monitoring application. If name is not specified, then by default will be used *Address*.

Address – (mandatory) the address of the object. This can be FQDN or IP.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Add host

Add-Host [-Name <String>] -Address <String> [-Login <String>] [-Password <String>]

Add host to monitoring list.

Name – (optional) the name of the object that will be displayed in monitoring application. If name is not specified, then by default will be used *Address*.

Address – (mandatory) the address of the object. This can be FQDN, IP or computer name.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Add port

Add-Port [-Name <String>] -Address <String> -Port <UInt16>

Add port on specified address to monitoring list.

Name – (optional) the name of the object that will be displayed in monitoring application. If name is not specified, then by default will be used combination *Address:Port*

Address – (mandatory) the address of the object. This can be FQDN, IP or computer name.



Port – (mandatory) the port number to be monitored.

Add service

Add-Service [-Name <String>] -Address <String> -ServiceName <String> [-Login <String>] [-Password <String>]

Add service on specified address to monitoring list.

Name – (optional) the name of the object that will be displayed in monitoring application. If name is not specified, then by default will be used *Address*.

Address – (mandatory) the address of the object. This can be FQDN, IP or computer name.

ServiceName – (mandatory) the name of the service to be monitored.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Set datacenter

Set-DataCenter -Id <Guid> [-Name <String>] [-Login <String>] [-Password <String>]

Update any properties of the datacenter.

Id – (mandatory) the unique identifier of the monitored object.

Address – (optional) the address of the object. This can be FQDN or IP.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Set cluster

Set-Cluster -Id <Guid> [-Name <String>] [-Address <String>] [-Login <String>] [-Password <String>]

Update any properties of the monitored cluster.

Id – (mandatory) the unique identifier of the monitored object.

Name – (optional) the name of the object that will be displayed in monitoring application.

Address – (optional) the address of the object. This can be FQDN or IP.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Set host

Set-Host -Id <Guid> [-Name <String>] [-Address <String>] [-Login <String>] [-Password <String>]

Update any properties of the monitored host.

Id – (mandatory) the unique identifier of the monitored object.

Name – (optional) the name of the object that will be displayed in monitoring application.

Address – (optional) the address of the object. This can be FQDN, IP or computer name.

Login – (optional) the username under which credentials the object will be queried by 5nine vMonitor service.

Password – (optional) the password of the username.

Set port

Set-Port -Id <Guid> [-Name <String>] [-Address <String>] [-Port <UInt16>]

Update any properties of the monitored port.

Id – (mandatory) the unique identifier of the monitored object.

Name – (optional) the name of the object that will be displayed in monitoring application.

Address – (optional) the address of the object. This can be FQDN, IP or computer name.

Port – (optional) the port number to be monitored.

Set service

Add-Service -Id <Guid> [-Name <String>] [-Address <String>] [-ServiceName <String>] [-Login <String>] [-Password <String>]

Update any properties of the monitored service.

Id – (mandatory) the unique identifier of the monitored object.

Name – (optional) the name of the object that will be displayed in monitoring application. If name is not specified, then by default will be used *Address*.

Address – (optional) the address of the object. This can be FQDN, IP or computer name.

ServiceName – (optional) the name of the service to be monitored.

Login – (optional) the username under which credentials the object will be queried by 5nine

vMonitor service.

Password – (optional) the password of the username.

Remove object

Remove-Object -Id <Guid>

Remove monitored object by its Id.

Id – (mandatory) the unique identifier of the monitored object.

Set object monitoring status

Set-MonitorStatus -Id <Guid> -Status <string>

Set monitoring status of the object.

Id – (mandatory) the unique identifier of the object.

Status – (mandatory) possible values are: Monitored, Paused. In 'Paused' state monitoring of the object is disabled.

Get monitored objects

Get-MonitoredObjects

Get list of all monitored objects, including affiliated nodes and guests.

Options commands

Get global options

Get-Options

Get global monitoring options.

Set global options

Set-Options [-Frequency <UInt32>] [-Targets <String>] [-AutoStartMonitoring <bool>] [-UserName <String>] [-Password <String>] [-WarningActions <String>] [-ErrorActions <String>] [-IsPolicyEvaluationEnabled <bool>] [-SmtpAddress <String>] [-SmtpPort <UInt16>] [-SmtpSSL <bool>] [-SmtpRequiresAuthentication <bool>] [-SmtpUserName <String>] [-SmtpPassword <String>] [-SmtpSender <String>] [-SmtpRecipients <String>]

Set any properties of the global monitoring options.

Frequency – (optional) the frequency of pulling data from monitored objects in seconds.



Targets – (optional) target types of monitoring. Possible values are: **None, TargetOnly, TargetAndGuests**

AutoStartMonitoring – (optional) specifies if monitoring should automatically start on service starting.

UserName – (optional) username under credentials of whom data pulling will be executed.

Password – (optional) the password of the username.

WarningActions – (optional) specifies action that should be performed on appearing of warning alert. Possible values are: **None, AddEvent, SendEmail**

ErrorActions – (optional) specifies action that should be performed on appearing of error alert. Possible values are: **None, AddEvent, SendEmail**

IsPolicyEvaluationEnabled – (optional) specifies if policy should be evaluated against set of alert definitions.

SmtpAddress – (optional) address of smtp server.

SmtpPort – (optional) port number of smtp server.

SmtpSSL – (optional) Is SSL enabled.

SmtpRequiresAuthentication – (optional) is smtp requires authentication.

SmtpUserName – (optional) username to logon smtp server.

SmtpPassword – (optional) password to logon smtp server.

SmtpSender – (optional) email address from which emails will be sent.

SmtpRecipients – (optional) email addresses to which alerts information will be sent.

Get indicators list

Get list of available indicators that collect workload data.

Get-Indicators

Get alert definitions

Get-AlertDefinition [-Id <Guid>]

Get alert definition(s).

Id – (optional) identifier of specific alert. If parameter not set, all alert definitions will be

returned.

Modify alert definition

Set-AlertDefinition [-Entity <AlertDefinitionPOCO>] [-Id <Guid>] [-Name <String>] [-Description <String>] [-IsEnabled <bool>] [-Condition <String>] [-WarningThreshold <float>] [-ErrorThreshold <float>] [-ConditionLength <int>] [-ReportFrequency <int>] [-ReportRange <float>]

Modify settings of alert definition.

Entity – (optional) the whole object of type FiveNine.Monitor.AlertDefinitionPOCO that must be updated.

Note: Instead of updating by the whole object you can use parameters to update any specific properties of alert definition by specifying its id.

Id – (optional) identifier of specific alert.

Name – (optional) name of alert.

Description – (optional) description of alert.

IsEnabled – (optional) is alert enabled.

Condition – (optional) operation of the condition that will be checked during policy evaluation.
Possible values: **Equals, Above, Below**

WarningThreshold – (optional) level of condition to trigger warning alert.

ErrorThreshold – (optional) level of condition to trigger error alert.

ConditionLength – (optional) time period in seconds under which condition must be true – to trigger alert.

ReportFrequency – (optional) specifies the frequency for alert reporting.

Set credentials

Set-Credentials [-EntityId <Guid>] -Login <String> -Password <String>

Set credentials for global monitoring options or for specified object.

EntityId – (optional) identifier of specific object. If not set then credentials will be set for global monitoring options.

Login – (optional) username.

Password – (optional) password.



Results commands

Get summary status

Get-SummaryStatus -EntityId <Guid>

Get summary information about the current status of monitored object.

EntityId – (mandatory) the unique identifier of the monitored object.

Get summary results

Get-SummaryResults -EntityId <Guid>

Get summary results on Memory, CPU, Disk and Network Usage for the aggregating object types as DataCenter or Cluster.

EntityId – (mandatory) the unique identifier of the monitored object.

Get triggered alerts

Get-TriggeredAlerts -EntityId <Guid>

Get triggered alerts for the monitored object.

EntityId – (mandatory) the unique identifier of the monitored object.

Remove triggered alert

Remove-TriggeredAlert -Id <Int32>

Remove triggered alert.

Id – (mandatory) the unique identifier of the monitored object.

Get workload data

Get-MonitorResults -EntityId <Guid> -IndicatorId <Guid> [-Start <DateTime>] [-End <DateTime>] [-Period <String>] [-Count <Int32>]

Get data collected during monitoring of the specified monitored object for specified indicator and given period of time.

EntityId – (mandatory) the unique identifier of the monitored object.

IndicatorId – (mandatory) the unique identifier of the indicator.

Start – (optional) date/time of the beginning of the period.

End – (optional) date/time of the ending of the period.

Period – (optional) base of the period. Possible values are: **RealTime, Hourly, Daily**. If parameter is not specified by default used **Hourly**.

Count – (optional) number of last results (in realtime period) that need to be returned. This parameter is ignored if Start or End parameter was specified.

Note: If all Start, End and Count are not specified then will be returned all results.

In the list below description of available indicators.

Indicator Id	Scope	Description
B3815B05-DAD9-42AC-B380-B6F7D87C8369	Host, Node	Number of Virtual Processors
E739568E-F268-4333-8E8B-C0EACC4B2B66	Host, Node	Number of Logical Processors
D37B491E-A687-4243-8A21-192768E1729F	Host, Node	Logical Processor Usage by Guests
C78DBF89-5929-4E74-A312-7B5BD772627F	Host, Node	Logical Processor Usage by Hypervisor
E2F89331-F3DC-4E5E-AFC5-3EA23C4AAF62	Host, Node	Virtual Processor Usage by Guests
C364E108-423B-46B9-BCA8-34130AA1C073	Host, Node	Virtual Processor Usage by Hypervisor
A2F8BEEE-2FC0-4B36-BC3C-7DE24281E18F	Host, Node	Memory Available, Mb
1BB4FB9A-D678-47E2-9860-7DFEFDC9E8AB	Host, Node	Total Physical Memory, Mb
E92CB28E-6B25-4601-82D6-D36F68F77FE2	Host, Node	Memory Pressure, Pages/sec
2F7CE8BE-5AEA-48E2-AB75-53A1FF0FA4C8	Host, Node	Network Interfaces Usage, Bytes/sec
BFD4282F-4719-4BED-9EA6-B69AEF0EDD12	Host, Node	Network Interfaces Bandwidth, Bits/sec
C62E1053-931C-4395-851B-E9805FFAE627	Host, Node	Network Interfaces Output Queue Length
A4FA05E4-DFD9-4C83-8B21-136D39D39E93	Host, Node	Physical Disks, Bytes/sec
EFD14BA1-5BA4-4F41-8097-C32BE6FF7431	Host, Node	Physical Disks Current Queue Length

Indicator Id	Scope	Description
BC15F9B6-E990-495F-9C86-4752A80A7AA1	Host, Node	Network Utilization, %
532CFC96-0AE5-4F75-A626-C0944C8EEA14	Host, Node	Disk IO Usage, %
74E3CE50-0562-4BFF-9CF8-B8681523DE14	Guest	Processor usage of the virtual system
BFF48244-80A8-4D70-B194-1CF603FC60D4	Guest	Virtual Machine Available Memory Buffer
9A94A354-3379-48B4-B3B8-D2D8AB851EE0	Guest	Virtual Network Utilization, Bytes/sec
291941BE-DA71-44E7-89B4-C1078181B97E	Guest	Virtual Disk Read, Bytes/sec
DC453B5D-FB9D-4AF4-B484-07B9289E1190	Guest	Virtual Disk Write, Bytes/sec

Misc commands

Get monitoring status

Get-MonitorStatus

Get status of monitoring. Return values: None, Monitored, Paused.

Start monitoring

Invoke-Start

Start monitoring.

Stop monitoring

Invoke-Stop

Stop monitoring.

Examples

- Load snapin to PowerShell console

Add-pssnapin vmonitorapi

- Set some options:

set-options -SmtpPort 465 -ErrorActions SendEmail -WarningAction AddEvent

- Add object to monitoring:

```
add-host -address hv1.crossinform.local
```

```
add-host -address hv2.crossinform.local -Name HV2 -Login crossinform\user -Password pa$$word
```

```
add-cluster -address hvcluster.crossinform.local -Name HVCLUSTER -Login crossinform\user -Password pa$$word
```

```
add-port -address neufor.crossinform.local -port 8080 -name TFSPort
```

```
add-service -address neufor.crossinform.local -servicename AdobeARMservice -login crossinform\user -Password pa$$word
```

- Get monitored objects

```
get-monitoredobjects
```

- Set parameter 'Name' to 'HV1' for host hv1.crossinform.local

```
set-host -id (get-monitoredobjects | where {$_.Address -eq "hv1.crossinform.local"}).Id -Name HV1
```

- Change port number

```
set-port -id (get-monitoredobjects | where {$_.Name -eq "TFSPort"}).Id -Port 8088
```

- Remove hv1.crossinform.local

```
remove-object -id (get-monitoredobjects | where {$_.Address -eq "hv1.crossinform.local"}).Id
```

- Get all rules

```
get-alertdefinition
```

- Get specific rule

```
get-alertdefinition -id b9d5e99a-1f20-476e-a71d-e5cfd2defbd5
```

- Change rule settings

```
set-alertdefinition -id b9d5e99a-1f20-476e-a71d-e5cfd2defbd5 -IsEnabled $true -WarningThreshold 85
```

- Get triggered alerts (with description) for monitored service AdobeARMservice

```
get-triggeredalerts -entityId (get-monitoredobjects | where {$_.Address -eq "neufor.crossinform.local:AdobeARMservice"}).Id | select Id, Severity, TriggeredTime, @{{Expression={(get-alertdefinition -Id $_.AlertId).Description};Label="Description"}} | Format-List
```



- Remove triggered alert

remove-triggeredalert -id 2237

- Get summary status of object

get-summarystatus -entityId (get-monitoredobjects | where {\$_.Address -eq "hv2.crossinform.local"}).Id

- Get workload data of hv2.crossinform.local for indicator HypervisorLogicalProcessors on hourly basis

get-monitorresults -entityId (get-monitoredobjects | where {\$_.Address -eq "hv2.crossinform.local"}).Id -indicatorId (get-indicators | where {\$_.Name -eq "HypervisorLogicalProcessors"}).Id -Period Hourly

- Set credentials for monitoring hvcluster.crossinform.local

set-credentials -entityId (get-monitoredobjects | where {\$_.Address -eq "hvcluster.crossinform.local"}).Id -Login crossinform\user -Password pa\$\$word

REST API

REST service is available by address <http://localhost:8734/FiveNine.vMonitor>

REST service supports both xml and json data formats. For requests REST service automatically determines which type of data is sent by analyzing contentType header ('application/json' or 'application/xml').

But to get returning data in preferable format you need specify it explicitly by adding parameter to query, for example: <http://localhost:8734/FiveNine.vMonitor/MonitoredEntities?format=json>

If format parameter is omitted, then by default will be used format=xml.

Authentication

LogOn

URI: <http://localhost:8734/FiveNine.vMonitor/LogOn>

Type: POST

Description: authenticate user - returns authentication ticket and set it as cookie in header response.

XML data format:

```
<LogOn xmlns="http://5nine.com/Monitor">
```



<login></login>

<password> </password>

</LogOn>

JSON data format:

{login: <string>, password: <string>}

Return data:

Guid

LogOff

URI: <http://localhost:8734/FiveNine.vMonitor/LogOff>

Type: POST

Description: close the session.

XML data format:

<empty/>

Set master credentials

URI: <http://localhost:8734/FiveNine.vMonitor/MasterCredentials>

Type: POST

Description: Set login/password for predefined user.

XML data format (example):

<SetMasterCredentials xmlns="http://5nine.com/Monitor">

<login></login>

<password> </password>

</ SetMasterCredentials >

JSON data format:

{login: <string>, password: <string>}



Monitoring objects

Add/Update objects

URI: <http://localhost:8734/FiveNine.vMonitor/MonitoredEntities>

Type: PUT

Description: add objects to monitoring list if object with such id is not found, otherwise update it.

XML data format:

```
<ArrayOfManagedEntity xmlns="http://5nine.com/Monitor">
<ManagedEntity><Address/> <Id/><Login/><Name/> <Password/><TypeId/></ManagedEntity>
</ArrayOfManagedEntity>
```

JSON data format:

```
[[{Address: <string>, Name:<string>, Login:<string>, Password: <string>, TypeId: <Guid>}]]
```

Note:

Possible TypeId values:

DataCenter: 02443586-D556-4784-9851-F44F7B4C979A

Cluster: 1709B140-35A5-46B7-8176-FF519FAE0244

Host: EED85E1E-ADD5-4EB2-AE0A-4EDA6508A650

Port: B08A7FAF-5C08-4CE2-A7C7-5287A41D99D7

Service: B39D2DE4-5AF8-4366-8366-F49AD98E632E

Set objects

URI: <http://localhost:8734/FiveNine.vMonitor/MonitoredEntities>

Type: POST

Description: set objects to monitoring list. All previous objects will be cleared.

XML data format:

```
<ArrayOfManagedEntity xmlns="http://5nine.com/Monitor">
<ManagedEntity><Address/> <Id/><Login/><Name/> <Password/><TypeId/></ManagedEntity>
```



</ArrayOfManagedEntity>

JSON data format:

`[[Address: <string>, Name:<string>, Login:<string>, Password: <string>, TypeId: <Guid>]]`

Remove object

URI: <http://localhost:8734/FiveNine.vMonitor/MonitoredEntities/{id}>

Type: DELETE

Description: remove specified object from monitoring.

Set object monitoring status

URI: <http://localhost:8734/FiveNine.vMonitor/MonitorStatus/{id}/{status}>

Type: POST

Description: Set monitoring status of the object.

Id – the unique identifier of the object.

Status – possible values are: Monitored, Paused. In ‘Paused’ state monitoring of the object is disabled.

Get monitored objects

URI: <http://localhost:8734/FiveNine.vMonitor/MonitoredEntities/{id}>

Description: Get list of all monitored objects, including affiliated nodes and guests.

Type: GET

Options commands

Get global options

URI: <http://localhost:8734/FiveNine.vMonitor/Options>

Type: GET

Description: Get global monitoring options.

Set global options

URI: <http://localhost:8734/FiveNine.vMonitor/Options>

Type: PUT



Description: update global monitoring options.

XML data format (example):

```
<MonitorOptions xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://5nine.com/Monitor"><AutoStartMonitoring>true</AutoStartMonitoring><ErrorActions>AddEvent</ErrorActions><Frequency>10</Frequency><IsPolicyEvaluationEnabled>true</IsPolicyEvaluationEnabled><MailNotifications><Address/><Password
i:nil="true"/><Port>25</Port><Recipients/><RequiresAuthentication>false</RequiresAuthentication><SSL>false</SSL><Sender/><UserName>pppp</UserName></MailNotifications><Password
i:nil="true"/><Targets>TargetAndGuests</Targets><UserName
i:nil="true"/><WarningActions>None</WarningActions></MonitorOptions>
```

JSON data format (example):

```
{ "AutoStartMonitoring": true, "ErrorActions": 1, "Frequency": 10, "IsPolicyEvaluationEnabled": true, "MailNot
ifications": { "Address": "", "Password": null, "Port": 25, "Recipients": "", "RequiresAuthentication": false, "SSL": f
alse, "Sender": "", "UserName": "pppp" }, "Password": null, "Targets": 2, "UserName": null, "WarningActions": 0
}
```

Get alert definitions

URI: <http://localhost:8734/FiveNine.vMonitor/AlertDefinition>

Type: GET

Description: Get alert definitions.

Modify alert definition

URI: <http://localhost:8734/FiveNine.vMonitor/AlertDefinition>

Type: PUT

Description: Modify settings of alert definition.

XML data format (example):

```
<AlertDefinitionPOCO xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://5nine.com/Monitor"><Category>2</Category><Condition>1</Condition><ConditionLength
h>300</ConditionLength><Description>A condition when the percentage of time spent by the processor
in guest and hypervisor code is greater than specified
threshold.</Description><ErrorThreshold>95</ErrorThreshold><Id>e56882ae-0c07-485f-9c99-071e69e8
580f</Id><IsEnabled>true</IsEnabled><Name>Logical
Processor</Name><ReportFrequency>0</ReportFrequency><ReportRange>0</ReportRange><Scope>1<
/Scope><TriggerRule>% Total Run
Time</TriggerRule><UOM>0</UOM><WarningThreshold>90</WarningThreshold></AlertDefinitionPOC
O>
```



JSON data format (example):

```
{ "Category":2, "Condition":1, "ConditionLength":300, "Description":"A condition when the percentage of time spent by the processor in guest and hypervisor code is greater than specified threshold.", "ErrorThreshold":95, "Id":"e56882ae-0c07-485f-9c99-071e69e8580f", "IsEnabled":true, "Name":"Logical Processor", "ReportFrequency":0, "ReportRange":0, "Scope":1, "TriggerRule":"% Total Run Time", "UOM":0, "WarningThreshold":90 }
```

Set credentials

URI: <http://localhost:8734/FiveNine.vMonitor/Credentials/{id}>

Type: *PUT*

Description: Set credentials for global monitoring options or for specified object.

XML data format (example):

```
<SetCredentials xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://5nine.com/Monitor"><login>crossinform\user</login><password>pa$$word</password></SetCredentials>
```

JSON data format (example):

```
{ "login":"crossinform\user", "Password":"pa$$word" }
```

Results commands

Get summary status

URI: <http://localhost:8734/FiveNine.vMonitor/SummaryStatus/{id}>

Type: *GET*

Description: Get summary status of specified object.

Get summary results

URI: <http://localhost:8734/FiveNine.vMonitor/SummaryResults/{id}>

Type: *GET*

Description: Get summary workload results of the specified object. Applied only to Cluster and Datacenter object types.

Get triggered alerts

URI: <http://localhost:8734/FiveNine.vMonitor/TriggeredAlert/{id}>

Type: *GET*



Description: Get triggered alert(s) for the monitored object. If {Id} is omitted then will be returned all triggered alerts.

Remove triggered alert

URI: <http://localhost:8734/FiveNine.vMonitor/TriggeredAlert/{id}>

Type: DELETE

Description: Remove triggered alert.

Get workload data

URI:

<http://localhost:8734/FiveNine.vMonitor/MonitorResults/{entityId}/{indicatorId}/{from}/{to}/{interval}>

Type: GET

Description: Get data collected during monitoring of the specified object for specified indicator and for given period of time.

Format of dates (from and to): yyyy-MM-ddTHH-mm-ss

Interval: RealTime, Hourly, Daily

URI: <http://localhost:8734/FiveNine.vMonitor/MonitorResults/{entityId}/{indicatorId}/{count}>

Type: GET

Description: Get last {count} samples of data collected during monitoring of the specified object for specified indicator.

Misc commands

Get monitoring status

URI: <http://localhost:8734/FiveNine.vMonitor/MonitorStatus>

Type: GET

Description: Get status of monitoring. Return values: None, Monitored, Paused.

Start monitoring

URI: <http://localhost:8734/FiveNine.vMonitor/Start>

Type: POST

Description: Start monitoring



XML data format (example): <empty/>

Stop monitoring

URI: <http://localhost:8734/FiveNine.vMonitor/Stop>

Type: POST

Description: Stop monitoring

XML data format (example): <empty/>

Examples

Examples below based on jquery and jquery rest extension. You also can browse these directly in folder "Program Files/5nine/5nine Hyper-V Cloud vMonitor/Web".

- Add object to monitoring:

```
$.put({
    contentType: 'application/json',
    url: restUri() + 'MonitoredEntities',
    data: JSON.stringify([
        { Address: $("input[name='Address']").val(), Name:
        $("input[name='Name']").val(), Login: $("input[name='Login']").val(), Password:
        $("input[name='Password']").val(), "TypeId": "eed85e1e-add5-4eb2-ae0a-4eda6508a650" }
    ]),
    complete: function () {
        InitializeTree();
    }
});
```

- Remove object

```
$.del({
    contentType: 'application/json',
    url: restUri() + 'MonitoredEntities/' + itemid,
    complete: function () {
        InitializeTree();
    }
});
```

- Set credentials

```
$.put({
    contentType: 'application/json',
    url: restUri() + 'Credentials/' + itemid,
    data: JSON.stringify( { login: $("input[name='Login']").val(), password:
$("input[name='Password']").val()})
});
```

- Modify settings

```
$.put({
    contentType: 'application/json',
    url: restUri() + 'Options',
    data: JSON.stringify(data)
});
```

OData Feed

OData Feed is available by address <http://localhost:8735/FiveNine.vMonitor>

To use OData feed requestor must be authenticated by REST service and request header must have TicketId cookie, otherwise Access Denied error will be returned in XML.

OData Feed also supports json/jsonp data formats.

To set json format add parameter \$format to URI:

[http://localhost:8735/FiveNine.vMonitor/MonitoredEntities?\\$format=json](http://localhost:8735/FiveNine.vMonitor/MonitoredEntities?$format=json)

To set jsonp format add parameter \$format and \$callback to URI:

[http://localhost:8735/FiveNine.vMonitor/MonitoredEntities?\\$format=json&\\$callback=funcCallback](http://localhost:8735/FiveNine.vMonitor/MonitoredEntities?$format=json&$callback=funcCallback)

Entities

List of monitored objects

URI: <http://localhost:8735/FiveNine.vMonitor/MonitoredEntities>



List of triggered alerts

URI: <http://localhost:8735/FiveNine.vMonitor/TriggeredAlerts>

List of alert definitions

URI: <http://localhost:8735/FiveNine.vMonitor/AlertDefinitions>