

# VMware ESXi Audit Reference

VMware vSphere 8.0

VMware ESXi 8.0b

vCenter Server 8.0

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

Copyright © 2023 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

# Contents

	About This Reference	8
<b>1</b>	account.changePassword	9
<b>2</b>	account.create	10
<b>3</b>	account.delete	11
<b>4</b>	account.edit	12
<b>5</b>	account.locked	13
<b>6</b>	api.call	14
<b>7</b>	audit.failure	15
<b>8</b>	audit.fetch	16
<b>9</b>	audit.net.failure	17
<b>10</b>	audit.storage.failure	18
<b>11</b>	audit.storage.recycle	19
<b>12</b>	audit.start	20
<b>13</b>	audit.stop	21
<b>14</b>	cgi.connect	22
<b>15</b>	cgi.disconnect	23
<b>16</b>	dcui.login	24
<b>17</b>	dcui.logout	25
<b>18</b>	entropy.rbg.failure	26

- 19** entropy.available.low 27
- 20** entropy.failure 28
- 21** entropy.link 29
- 22** entropy.start 30
- 23** entropy.stop 31
- 24** invalid.message 32
- 25** network.fw.defaultpolicy 33
- 26** network.fw.refresh 34
- 27** network.fw.rule.disable 35
- 28** network.fw.rule.enable 36
- 29** network.fw.rule.update 37
- 30** network.nic.connect 38
- 31** network.nic.disconnect 39
- 32** network.switch.add 40
- 33** network.switch.remove 41
- 34** product.issue 42
- 35** proxy.connect 43
- 36** proxy.disconnect 44
- 37** service.ticket.get 45
- 38** service.ticket.use 46
- 39** service.token.get 47

- 40** service.token.use 48
- 41** settings.advanced.add 49
- 42** settings.advanced.delete 50
- 43** settings.advanced.reset 51
- 44** settings.advanced.set 52
- 45** settings.kernel.set 53
- 46** shell.cmd 54
- 47** ssh.cmd 55
- 48** ssh.connect 56
- 49** ssh.disconnect 57
- 50** ssh.session.begin 58
- 51** ssh.session.end 59
- 52** syslog.mark 60
- 53** syslog.net.connect 61
- 54** syslog.net.disconnect 62
- 55** syslog.net.link 63
- 56** syslog.reload 64
- 57** system.update.add 65
- 58** system.update.end 66
- 59** system.update.remove 67
- 60** system.update.start 68

- 61** time.parameters 69
- 62** usb.passthru.disable 71
- 63** usb.passthru.enable 72
- 64** vapi.connect 73
- 65** vapi.disconnect 74
- 66** vim.access.denied 75
- 67** vim.connect 76
- 68** vim.disconnect 77
- 69** vim.terminate 78
- 70** vm.create 79
- 71** vm.delete 80
- 72** vm.guestrpc.denied 81
- 73** vm.hypercall.denied 82
- 74** vm.kill 83
- 75** vm.net.add 84
- 76** vm.net.edit 85
- 77** vm.net.remove 86
- 78** vm.power.off 87
- 79** vm.power.on 88
- 80** vm.power.reset 89
- 81** vm.power.suspend 90

- 82** vm.reconfigure 91
- 83** vm.register 92
- 84** vm.snapshot.create 93
- 85** vm.snapshot.remove 94
- 86** vm.snapshot.removeAll 95
- 87** vm.snapshot.rename 96
- 88** vm.snapshot.revert 97
- 89** vm.storage.add 98
- 90** vm.storage.edit 99
- 91** vm.storage.remove 100
- 92** vm.unregister 101
- 93** vm.usb.connect 102
- 94** vm.usb.disconnect 103
- 95** x509.cacert.add 104
- 96** x509.cacert.remove 105
- 97** x509.srvcert.generate 106
- 98** x509.srvcert.install 107

# About This Reference

The *VMware ESXi Audit Reference* provides information about the audit records that ESXi generates in response to various security-related system events. The documentation includes the parameters associated with each event. You can access audit records through local storage or remote transmission. Audit record local storage and remote transmission are optional capabilities of ESXi.

## Intended Audience

This information is for experienced system administrators who are familiar with virtual machine technology and data center operations.

## Presence of Parameters

Each parameter's entry in an audit event's parameter table indicates whether the parameter is always present or is optional, and if applicable, under what conditions.

## Event Severity

Unless an audit event specifies otherwise, the severity of the audit event is `Informational` when the `result` value is `success` and `Notice` when the `result` value is `failure`.

## Document References

For more information on audit records, see Subsection "Audit Records" under [Log File Formats](#) in the [VMware ESXi Installation and Setup](#) guide. For more information on configuring local audit record storage and remote transmission, see [Configuring System Logging](#), also in the [VMware ESXi Installation and Setup](#) guide.



# account.changePassword

1

This event is generated when an attempt was made to change the password of a user account.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Optional
object	Specifies the name of the target account.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional
comment	Provides additional information about the event.	Optional

## account.create

2

This event is generated when an attempt was made to create a user account.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the new account.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## account.delete

3

This event is generated when an attempt was made to delete a user account.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the target account.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## account.edit

# 4

This event is generated when an attempt was made to edit an existing user account.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the target account.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code> .
ip	Specifies the network address of the originator of the operation	Always
opID	Specifies the operation ID.	Optional

# account.locked

5

This event is generated when an account was locked due to too many failed login attempts.

**Note** For `account.locked` the event severity is always `Warning`.

Parameter	Description	Presence
subject	Specifies the name of the user whose account was locked.	Always
object	The value is always <code>account</code> .	Always
result	The value is always <code>success</code> .	Always

This event is generated when an attempt was made to call an API.

**Note** For `api.call` the event severity when the `result` value is `success` is `Debug`.

Parameter	Description	Presence
<code>subject</code>	Specifies the name of the user who requested the operation. This might be an empty string for anonymous operations (operations that have the <code>System.anonymous</code> required privilege) or for operations performed by the system.	Always
<code>object</code>	Specifies the name of the object and the method on which the operation was invoked (for example <code>vmodl.query.PropertyCollector.createFilter</code> , <code>vim.EnvironmentBrowser.queryConfigOptionDescriptor</code> , or <code>vim.VirtualMachine.powerOn</code> ). Consult the vSphere Web Services API documentation for more information. Note that some objects are undocumented since they are internal-use only.	Always
<code>result</code>	The value is either <code>success</code> or <code>failure</code> .	Always

# audit.failure

7

This event is generated upon audit daemon restart after the audit daemon suffered a catastrophic failure. This indicates that a loss of audit records might have occurred.

**Note** For `audit.failure` the event severity is always `Alert`.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>audit</code> .	Always
result	The value is always <code>failure</code> .	Always
reason	The value is always potential loss of audit records due to daemon crash.	Always

## audit.fetch

# 8

This event is generated when an attempt was made to fetch audit records.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>audit</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always
opID	Specifies the operation ID.	Optional



# audit.net.failure

9

This event is generated when remote host connectivity was lost. This indicates that a loss of audit records might have occurred.

**Note** For `audit.net.failure` the event severity is always `Alert`.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the affected remote host.	Always
result	The value is always <code>failure</code> .	Always
reason	Specifies the reason for the network failure.	Always

# audit.storage.failure

10

This event is generated when an I/O error occurred while writing audit record storage. This indicates that a loss of audit records might have occurred.

**Note** For `audit.storage.failure` the event severity is always `Alert`.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the audit record storage file where the I/O error occurred.	Always
result	The value is always <code>failure</code> .	Always
reason	The value is always <code>I/O error</code> .	Always

## audit.storage.recycle

11

This event is generated when the audit record storage FIFO returned to its beginning (and older records are dropped).

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>audit</code> .	Always
result	The value is always <code>success</code> .	Always

## audit.start

12

This event is generated when audit record storage and/or transmission was started.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>audit</code> .	Always
result	The value is always <code>success</code> .	Always

## audit.stop

13

This event is generated when audit record storage and/or transmission was stopped.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>audit</code> .	Always
result	The value is always <code>success</code> .	Always

This event is generated when an attempt was made to establish a CGI connection.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>cgi</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always
url	Specifies the URL of the authentication request.	Always
http.method	Specifies the HTTP method involved in the connection attempt.	Always

## cgi.disconnect

15

This event is generated when a CGI connection was ended (explicit or time out).

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>cgi</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always
url	Specifies the URL of the authentication request.	Always
http.method	Specifies the HTTP method involved in the connection attempt.	Always

## dcui.login

16

This event is generated when an attempt was made to log in via the DCUI.

Parameter	Description	Presence
subject	Specifies the name of the user who attempted to log in.	Always
object	The value is always <code>dcui</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>



## dcui.logout

17

This event is generated when a DCUI login was ended (explicit or time out).

Parameter	Description	Presence
subject	Specifies the name of the user associated with the login.	Always
object	The value is always <code>dcui</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>

# entropy.rbg.failure

18

This event is generated when the host was unable to provide a sufficient amount of entropy for cryptographically secure random number generation.

**Note** For `entropy.rbg.failure` the event severity is always `Alert`.

Parameter	Description	Presence
<code>subject</code>	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
<code>object</code>	The value is always <code>rbg</code> .	Always
<code>result</code>	The value is always <code>failure</code> .	Always
<code>reason</code>	Indicates the reason for the failure. When testing for entropy failure, the value is <code>test</code> .	Always

## entropy.available.low

19

This event is generated when the amount of available entropy data has crossed the boundary of the low water mark (either going above or going below).

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user)	Always
object	The value is always <code>entropy</code> .	Always
result	The value is either <code>success</code> (has gone above) or <code>failure</code> (has gone below).	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>

## entropy.failure

20

This event is generated upon entropyd daemon restart after the daemon suffered a catastrophic failure (for example after the daemon or ESXi crashed). It indicates that the daemon was not shut down quiescently.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user)	Always
object	The value is always <code>entropy</code> .	Always
result	The value is always <code>failure</code> .	Always
reason	Indicates the reason for the failure.	Always

This event is generated when the connection to the external entropy source has been established (or restored), or broken. A connection is considered established (or restored) when ESXi receives entropy data from the external entropy source. A connection is considered broken when ESXi has not received entropy from the external entropy source within the configured timeout period.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user)	Always
object	The value is always <code>entropy</code> .	Always
result	The value is either <code>success</code> (connection established or restored) or <code>failure</code> (connection broken).	Always
reason	Indicates the reason for the failure.	When the <code>result</code> value is <code>failure</code>

## entropy.start

22

This event is generated when the entropy daemon starts.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user)	Always
object	The value is always <code>entropy</code> .	Always
result	The value is always <code>success</code> .	Always

## entropy.stop

23

This event is generated when the entropy daemon stopped.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user)	Always
object	The value is always <code>entropy</code> .	Always
result	The value is always <code>success</code> .	Always

## invalid.message

24

This event is generated when an event and its parameters prove problematic. Parameters from the problematic event form the body of this event to make it easier to determine how the problematic event was introduced.

**Note** For `invalid.message` the event severity is always `Error`.

Parameter	Description	Presence
<code>eventID</code>	Specifies the event ID of the problematic event.	Optional
Invalid parameter names	These are replaced with the name <code>badParam</code> .	Optional
Invalid parameter values	These are replaced with an empty string. (This can also affect <code>subject</code> and <code>object</code> parameter values.)	Optional
<code>result</code>	The value is always <code>failure</code> .	Always
<code>reason</code>	Indicates the reason(s) for the failure. (More than one may be specified.)	Optional



## network.fw.defaultpolicy

25

This event is generated when an attempt was made to update the firewall default policy.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>firewall</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional

## network.fw.refresh

26

This event is generated when an attempt was made to refresh the firewall.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>firewall</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>reason</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional

## network.fw.rule.disable

27

This event is generated when an attempt was made to disable a firewall ruleset.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the affected firewall ruleset.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional

## network.fw.rule.enable

28

This event is generated when an attempt was made to enable a firewall ruleset.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the affected firewall ruleset.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional

## network.fw.rule.update

29

This event is generated when an attempt was made to update a firewall ruleset.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the affected firewall ruleset.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional

## network.nic.connect

30

This event is generated when an attempt was made to connect a host's NIC to a network switch.

Parameter	Description	Presence
subject	Specifies the name of user who requested the operation.	Always
object	Specifies the name of the affected physical NIC.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional
type	The value is either <code>physical</code> or <code>vmkernel</code> .	Always
deviceID	Specifies the name of the affected vSphere Standard Switch or the uuid of the affected Distributed Virtual Switch.	Always

# network.nic.disconnect

31

This event is generated when an attempt was made to disconnect a host's NIC from a network switch.

Parameter	Description	Presence
subject	Specifies the name of user who requested the operation.	Always
object	Specifies the name of the affected physical NIC.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
opID	Specifies the operation ID.	Optional
type	The value is either <code>physical</code> or <code>vmkernel</code> .	Always
deviceID	Specifies the name of the affected vSphere Standard Switch or the uuid of the affected Distributed Virtual Switch.	Always

## network.switch.add

32

This event is generated when an attempt was made to add a virtual switch to the host.

Parameter	Description	Presence
subject	Specifies the name of user who requested the operation.	Always
object	Specifies the name of the virtual switch.	Always
type	The value is either <code>standard</code> or <code>distributed</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
opID	Specifies the operation ID.	Optional



## network.switch.remove

33

This event is generated when an attempt was made to remove a virtual switch from the host.

Parameter	Description	Presence
subject	Specifies the name of user who requested the operation.	Always
object	Specifies the name of the virtual switch.	Always
type	The value is either <code>standard</code> or <code>distributed</code> .	Always
result	The value is <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
opID	Specifies the operation ID.	Optional

This event is generated when a product detects that an audit record should be issued but it is unable to do so for some reason. It is also generated for system issues that could put the system into an inconsistent or undesirable state.

**Note** For `product.issue` the event severity is always `Alert`.

Parameter	Description	Presence
<code>subject</code>	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
<code>object</code>	Specifies the name of the affected product (for example <code>esx</code> ).	Always
<code>result</code>	The value is always <code>failure</code> .	Always
<code>reason</code>	Specifies the reason why the the audit record could not be emitted, or the reason for the system failure.	Always
<code>location</code>	Provides information about where in the product the issue occurred (for example a short description of the subsystem reporting the issue, like <code>VM state management</code> ).	Optional

This event is generated when an attempt was made to establish a proxy service connection.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>proxy</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Optional
port	Specifies the network port number of the originator of the connection.	Optional

## proxy.disconnect

36

This event is generated when a proxy service connection was ended.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	The value is always <code>proxy</code> .	Always
result	The value is always <code>success</code> .	Always
ip	Specifies the network address of the originator of the connection.	Optional
port	Specifies the network port number of the originator of the connection.	Optional

## service.ticket.get

37

This event is generated when a service authorization ticket was requested.

This `service.ticket.get` event is generated for the following service URL paths:

- `/cgi-bin`
- `/folder`
- `/host`
- `/screen`
- `/tmp`

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the ticket ID with most of its fields masked out.	Always
result	The value is always <code>success</code> .	Always
ip	Specifies the network address of the originator of the operation.	Always
url	Specifies the URL the ticket was acquired for.	Always
http.method	Specifies the HTTP method the ticket was acquired for, or empty string if not specified.	Always
opID	Specifies the operation ID.	Optional

## service.ticket.use

# 38

This event is generated when an attempt was made to use a service authorization ticket for the service URL paths listed under `service.ticket.get`. For more information, see [Chapter 37 service.ticket.get](#).

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation if the ticket is valid and is an empty string otherwise.	Always
object	Specifies the ticket ID with most of its fields masked out.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
url	Specifies the URL path of the incoming request.	Always
http.method	Specifies the HTTP method of the incoming request.	Always
opID	Specifies the operation ID.	Optional

## service.token.get

39

This event is generated when an attempt was made to request an ESXi access token from user credentials or another security token.

The `service.token.get` event is generated for the following service URL paths:

- `/cgi-bin`
- `/api`
- `/sdk`
- `/observability`
- `/gdp`
- `/stats`

Description	Parameter	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the signature of the token. The value is an empty string if the signature could not be determined.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## service.token.use

# 40

This event is generated when an attempt was made to execute an operation by passing a previously created access token for the service URL paths listed under `service.token.get`. For more information, see [Chapter 39 service.token.get](#).

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the signature of the token. The value is an empty string if the signature could not be determined.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional



## settings.advanced.add

41

This event is generated when an attempt was made to add an advanced option.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the advanced option.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
type	The value is either <code>int</code> or <code>string</code> .	Always
min	Denotes the minimum legal value.	When the <code>type</code> is an integer ( <code>int</code> )
max	Denotes the maximum legal value.	When the <code>type</code> is an integer ( <code>int</code> )
defaultValue	Specifies the default value of the new option.	Always
description	Specifies the description of the new option.	Always
isHidden	Indicates whether the option is hidden (value of 1) or not hidden (value of 0).	Always

## settings.advanced.delete

42

This event is generated when an attempt was made to delete an advanced option.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the advanced option.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always

## settings.advanced.reset

43

This event is generated when an attempt was made to reset (to default) an advanced option.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the advanced option.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
value	Specifies the default value for the option.	Optional

## settings.advanced.set

44

This event is generated when an attempt was made to change an advanced option.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the advanced option.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
value	Specifies the new value for the advanced option.	Always
opID	Specifies the operation ID.	Optional

## settings.kernel.set

45

This event is generated when an attempt was made to change a kernel setting.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the name of the setting.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
value	Specifies the new value for the kernel setting.	Always

## shell.cmd

# 46

This event is generated whenever a command, issued via a shell, completes.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>shell</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> , representing 0 or not 0, respectively, as indicated by the shell exit code.	Always
command	Specifies the command line.	Always
status	Specifies the exit code of the command.	When the <code>result</code> specifies <code>failure</code>

## ssh.cmd

47

This event is generated whenever a shell command was issued by `sshd`.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>ssh</code> .	Always
result	The value is always <code>success</code> , since the command is always passed successfully to the shell.	Always
command	Specifies the command passed to the shell.	Always

## ssh.connect

48

This event is generated when an attempt was made to begin an SSH connection.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>ssh</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always



## ssh.disconnect

49

This event is generated when an attempt was made to end an SSH connection.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>ssh</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always

## ssh.session.begin

50

This event is generated when an attempt was made to begin an SSH session.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>ssh</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the session.	Always

## ssh.session.end

51

This event is generated when an attempt was made to end an SSH session.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>ssh</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the session.	Always

## syslog.mark

52

This event is generated when a mark message is issued to syslog and the audit trail.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>syslog</code> .	Always
result	The value is always <code>success</code> .	Always
comment	Specifies the event message. The value can be an empty string.	Always

This event is generated when the ESXi syslog daemon, vmsyslogd, establishes a connection (UDP, TCP, or TLS (SSL)) to a remote host.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the remote host (a syslog collector) identifier (for example a DNS name, or an IPV4 or IPV6 address).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
subjectDN	Specifies the subject of the presented peer certificate in Distinguished Name (DN) form, for example <code>C=US,L=Palo Alto,O=VMware</code> .	When the <code>result</code> specifies <code>failure</code> and the root cause of the failure was due to a TLS (SSL) connection failure.
subjectAltName	Specifies the subject alternate name extension of the presented peer certificate, for example <code>IP:192.0.2.1</code> .	When the <code>result</code> specifies <code>failure</code> and the root cause of the failure was due to a TLS (SSL) connection failure.
referenceID	Specifies the address of the remote server (DNS, IPv4, or IPv6) from the client configuration. This is the reference identifier that is matched with the presented identifier.	When the <code>result</code> specifies <code>failure</code> and the root cause of the failure was due to a TLS (SSL) connection failure.

## syslog.net.disconnect

54

This event is generated when the ESXi syslog daemon, vmsyslogd, terminates a connection (UDP, TCP, or TLS (SSL)) with a remote host.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the remote host (a syslog collector) identifier (for example a DNS name, or an IPV4 or IPV6 address).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional

This event is generated when a remote host network connection changes state.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the identifier of the remote host (a syslog collector), for example a DNS name, or an IPV4 or IPV6 address.	Always
result	When the link transitions from up to down, the <code>result</code> value is <code>failure</code> . When the link transitions from down to up, the <code>result</code> value is <code>success</code> .	Always
reason	When the link transitions from up to down, the <code>reason</code> value is <code>link down</code> . When the link transitions from down to up, the <code>reason</code> value is <code>link up</code> .	Always

## syslog.reload

56

This event is generated when an attempt was made to force the syslog/audit daemon to reload its parameters.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>syslog</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always



# system.update.add

57

This event is generated upon completion of a VIB installation attempt.

**Note** For `system.update.add`, the event severity when the `result` value is `success` depends on whether an attempt was made to override signature validation. If such an attempt was made then for `success` the event severity is `Notice`. Otherwise it is the default (`Informational`). On the other hand, if the `result` value is `failure` then the event severity is always `Critical`.

Parameter	Description	Presence
<code>subject</code>	Specifies the name of the user who requested the operation.	Always
<code>object</code>	Specifies the VIB being installed.	Always
<code>result</code>	The value is either <code>success</code> or <code>failure</code> . If the value is <code>failure</code> , the reason is indicated by the <code>system.update.end</code> event. See <a href="#">Chapter 58 system.update.end</a> for more information.	Always

# system.update.end

58

This event is generated at the end of a system update.

**Note** For `system.update.end`, the event severity when the `result` value is `success` depends on whether an attempt was made to override signature validation. If such an attempt was made then for `success` the event severity is `Notice`. Otherwise it is the default (`Informational`). On the other hand, if the `result` value is `failure` then the event severity is always `Critical`.

Parameter	Description	Presence
<code>subject</code>	Specifies the name of the user who requested the operation.	Always
<code>object</code>	The value is always <code>system</code> .	Always
<code>result</code>	The value is either <code>success</code> or <code>failure</code> .	Always
<code>reason</code>	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>

# system.update.remove

59

This event is generated upon completion of a VIB removal attempt.

**Note** For `system.update.remove`, the event severity when the `result` value is `success` depends on whether an attempt was made to override signature validation. If such an attempt was made then for `success` the event severity is `Notice`. Otherwise it is the default (`Informational`). On the other hand, if the `result` value is `failure` then the event severity is always `Critical`.

Parameter	Description	Presence
<code>subject</code>	Specifies the name of the user who requested the operation.	Always
<code>object</code>	Specifies the VIB being removed.	Always
<code>result</code>	The value is either <code>success</code> or <code>failure</code> . If the value is <code>failure</code> , the reason is indicated by the <code>system.update.end</code> event. See <a href="#">Chapter 58 system.update.end</a> for more information.	Always

# system.update.start

60

This event is generated when an attempt was made to initiate a system update.

**Note** For `system.update.start`, the event severity when the `result` value is `success` depends on whether an attempt was made to override signature validation. If such an attempt was made then for `success` the event severity is `Notice`. Otherwise it is the default (`Informational`). On the other hand, if the `result` value is `failure` then the event severity is always `Critical`.

Parameter	Description	Presence
<code>subject</code>	Specifies the name of the user who requested the operation.	Always
<code>object</code>	The value is always <code>system</code> .	Always
<code>result</code>	The value is either <code>success</code> or <code>failure</code> .	Always
<code>reason</code>	Provides additional details about the event.	Optional

This event is generated when an attempt was made to change an NTP/PTP (clock synchronization) configuration parameter.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always an empty string (representing the host itself).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
operation	<p>Specifies the setting that is being affected.</p> <ul style="list-style-type: none"> <li>■ <code>ConfigureSettings</code>: Configure time settings. (TIME)</li> <li>■ <code>DaemonStateChange</code>: Values for this operation include <code>start</code>, <code>stop</code> and <code>restart</code>. (NTP, PTP)</li> <li>■ <code>LoadConfig</code>: Load a new NTP configuration file (NTP)</li> <li>■ <code>ResetToFactory</code>: Reset to factory settings (NTP, PTP)</li> <li>■ <code>SetDevice</code>: Set the network device type (PTP)</li> <li>■ <code>SetDomain</code>: Set the domain (PTP)</li> <li>■ <code>SetEnabled</code>: Enable or disable the service (NTP,PTP)</li> <li>■ <code>SetEvent</code>: Enable sending events to VC (TIME)</li> <li>■ <code>SetFallback</code>: Enable fallback to NTP from PTP</li> <li>■ <code>SetIpConfig</code>: Set the IP configuration (PTP)</li> <li>■ <code>SetLogLevel</code>: Set the logging level (NTP,PTP)</li> <li>■ <code>SetPort</code>: Set the port (PTP)</li> <li>■ <code>SetServers</code>: Set the list of servers or none if no servers (NTP)</li> <li>■ <code>UpdateConfig</code>: Update the time configuration (TIME, PTP, NTP)</li> <li>■ <code>UpdateFirewallRule</code>: Update the firewall rule (NTP, PTP)</li> <li>■ <code>ValidatePciPassthruDev</code>: Validate the pic passthru device (PTP)</li> </ul>	Always

Parameter	Description	Presence
backup	Specifies the use of NTP as fallback upon a PTP service failure.	Optional
comment	Provides information about the event.	Optional
deviceID	Specifies the type of the device in PTP.	Optional
deviceLabel	Specifies the PTP port device name.	Optional
domain	specifies the domain of PTP.	Optional
enabled	Provides information that pertains to enabled/disabled setting.	Optional
event	Specifies the time service that was sending events when failure was detected.	Optional
index	Specifies the port's index in PTP.	Optional
level	Specifies the logging level.	Optional
netmask	Specifies the subnet mask.	Optional
path	Specifies the path of an NTP configuration file.	Optional
protocol	The value is one of <code>DHCP</code> , <code>static IP address</code> , or <code>none</code> if no protocol applies.	Optional
servers	Specifies a list of servers.	Optional
type	Indicates the type of operation ( <code>TIME</code> for non-daemon settings, or <code>PTP</code> or <code>NTP</code> for daemon-specific settings).	Optional
value	Specifies the operation value.	Always

## usb.passthru.disable

62

This event is generated when an attempt was made to make a connected USB device unavailable for passthrough to a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies a reference to the affected USB device, in the form <code>bus:device:vendorID:productID</code> , where all values are represented in hex form.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Always
comment	Provides information about the event.	Optional

## usb.passthru.enable

63

This event is generated when an attempt was made to make a connected USB device available for passthrough to a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies a reference to the affected USB device, in the form <code>bus:device:vendorID:productID</code> , where all values are represented in hex form.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Always
comment	Provides information about the event.	Optional



This event is generated when an attempt was made to create a vAPI connection.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>vapi</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the connection.	Always
opID	Specifies the operation ID.	Optional

## vapi.disconnect

65

This event is generated when a vAPI connection was ended (explicit or time out).

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always <code>vapi</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## vim.access.denied

66

This event is generated when an attempt to access a protected VIM object was denied.

Parameter	Description	Presence
subject	Specifies the name of the user who was denied access.	Always
object	Specifies the name of the object on which the operation was invoked (for example <code>vim.host.StorageSystem:storageSystem</code> , <code>vmtoolsd.query.PropertyCollector:ha-property-collector</code> ). Consult the vSphere Web Services API documentation for more information. Note that some objects are undocumented since they are internal-use only.	Always
result	The value is always <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the connection.	Always
apiName	Describes the internal API name of the operation that was invoked (for example <code>GetStorageDeviceInfo</code> , <code>waitForUpdatesEx</code> ). To understand the meaning of this value, consult the vSphere Web Services API documentation, starting with the object value as a reference. Note that property retrieval operations are prefixed with <code>Is</code> for boolean properties and <code>Get</code> for all other properties. Also note that some operations are undocumented since they are internal-use only.	Always
path	Specifies the VMX file path if the target is a virtual machine, or the datastore name if the target object is a datastore.	Always
opID	Specifies the operation ID.	Optional

This event is generated when an attempt was made to create a VIM connection.

Parameter	Description	Presence
subject	Specifies the user authenticating with the system.	Always
object	The value is always <code>vim</code> .	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the connection.	Always
uniqueID	Specifies a unique ID representing the session that was established.	When the <code>result</code> specifies <code>success</code>
opID	Specifies the operation ID.	Optional

## vim.disconnect

68

This event is generated when a VIM connection was ended (explicit or time out).

Parameter	Description	Presence
subject	Specifies the owner of the connection.	Always
object	The value is always vim.	Always
result	The value is always success.	Always
reason	Is the reason for the connection ending, for example User Logout, Timeout expired, or Terminated by administrator.	Always
ip	Specifies the network address of the originator of the connection.	Always
uniqueID	Specifies a unique ID representing the closed session.	Always
opID	Specifies the operation ID.	Optional

## vim.terminate

69

This event is generated when a VIM connection was terminated by an administrator.

Parameter	Description	Presence
subject	Specifies the user who terminated the connection.	Always
object	Specifies the user who owned the terminated connection.	Always
result	The value is always <code>success</code> .	Always
ip	Specifies the network address of the originator of the connection.	Always
uniqueID	Specifies a unique ID representing the terminated session.	Always
opID	Specifies the operation ID.	Optional

## vm.create

70

This event is generated when an attempt was made to create a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## vm.delete

71

This event is generated when an attempt was made to delete a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional



## vm.guestrpc.denied

72

This event is generated when an attempt to use a privileged Guest RPC was denied.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is always <code>failure</code> .	Always
reason	The value is always <code>permission denied</code> .	Always
name	Specifies the Guest RPC command.	Always
value	Specifies the Guest RPC number.	Always

## vm.hypercall.denied

73

This event is generated when an attempt to use a hypercall was denied.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is always <code>failure</code> .	Always
reason	Indicates the reason for the denial.	Always
type	The value is either <code>Backdoor</code> or <code>HbBackdoor</code> .	Always
value	Specifies the hypercall number.	Always
name	Specifies the name of the hypercall.	Optional

This event is generated when an attempt was made to forcibly terminate a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always

## vm.net.add

75

This event is generated when an attempt was made to attach a network to a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user requesting the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
ip	Specifies the network address of the originator of the operation.	Always
networkID	Specifies the name of the network that was to be attached.	Always
opID	Specifies the operation ID.	Optional

This event is generated when an attempt was made to modify a network attached to a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
ip	Specifies the network address of the originator of the operation.	Always
networkID	Specifies the name of the new network.	Always
oldID	Specifies the name of the old network, or an empty string if not applicable.	Always
connect	The value is either <code>at power-on</code> or <code>not at power-on</code> when a change affects device setup.	Optional
opID	Specifies the operation ID.	Optional

## vm.net.remove

77

This event is generated when an attempt was made to detach a network from a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
ip	Specifies the network address of the originator of the operation.	Always
networkID	Specifies the name of the network that was to be detached.	Always
opID	Specifies the operation ID.	Optional

## vm.power.off

78

This event is generated when an attempt was made to power off a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user powering off a virtual machine.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional

This event is generated when an attempt was made to power on a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional



## vm.power.reset

80

This event is generated when an attempt was made to reset a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional

## vm.power.suspend

81

This event is generated when an attempt was made to suspend a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional

## vm.reconfigure

82

This event is generated when an attempt was made to reconfigure a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional

## vm.register

83

This event is generated when an attempt was made to register a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## vm.snapshot.create

84

This event is generated when an attempt was made to create a virtual machine snapshot.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
name	Specifies the display name of the snapshot.	Always
uniqueID	Specifies the unique ID of the snapshot. (This can be an empty string.)	Always
parent	Specifies the unique ID of the parent snapshot. If the snapshot is the root of the tree then the value is an empty string.	Always
opID	Specifies the operation ID.	Optional

## vm.snapshot.remove

85

This event is generated when an attempt was made to remove a virtual machine snapshot.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
name	Specifies the display name of the snapshot.	Always
uniqueID	Specifies the unique ID of the snapshot.	Always
opID	Specifies the operation ID.	Optional
comment	Provides information about the event.	Optional

## vm.snapshot.removeAll

86

This event is generated when an attempt was made to remove all virtual machine snapshots.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## vm.snapshot.rename

87

This event is generated when an attempt was made to rename a virtual machine snapshot.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
name	Specifies the current display name of the snapshot.	Always
newName	Specifies the new display name of the snapshot.	Always
uniqueID	Specifies the unique ID of the snapshot.	Always
opID	Specifies the operation ID.	Optional
comment	Provides information about the event.	Optional



## vm.snapshot.revert

88

This event is generated when an attempt was made to revert to virtual machine snapshot.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
name	Specifies the display name of the snapshot.	Always
uniqueID	Specifies the unique ID of the snapshot.	Always
opID	Specifies the operation ID.	Optional

## vm.storage.add

89

This event is generated when an attempt was made to add a virtual storage device to a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
path	Specifies the path to the affected file or an empty string when a local device is auto selected (for example CDROM/DVD).	Always
opID	Specifies the operation ID.	Optional

This event is generated when an attempt was made to modify the backing" of a virtual storage device associated with a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
path	Specifies the path to the attached file or an empty string when a local device is auto selected (for example CDROM/DVD).	Always
oldPath	Specifies the path to the detached file or an empty string when a local device is auto selected (for example CDROM/DVD).	Always
connect	The value is <code>at power-on</code> or <code>not at power-on</code> when a change affects device setup.	Optional
opID	Specifies the operation ID.	Optional

## vm.storage.remove

91

This event is generated when an attempt was made to remove a virtual storage device from a virtual machine.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
path	Specifies the path to the affected file or an empty string when a local device is auto selected (for example CDROM/DVD).	Always
opID	Specifies the operation ID.	Optional

## vm.unregister

92

This event is generated when an attempt was made to unregister a virtual machine.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the path of the VMX file of the affected virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Indicates the reason for the failure.	When the <code>result</code> specifies <code>failure</code>
ip	Specifies the network address of the originator of the operation.	Always
opID	Specifies the operation ID.	Optional

## vm.usb.connect

93

This event is generated when an attempt was made to connect a USB device to a virtual machine.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the path of the VMX file of the reporting virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
opID	Specifies the operation ID.	Optional
deviceLabel	The value has the form <code>usb:&lt;N&gt;</code> for usb 1.1, <code>ehci:&lt;N&gt;</code> for usb 2.0, or <code>usb_xhci:&lt;N&gt;</code> for usb 3.x. The <code>&lt;N&gt;</code> is a small integer, uniquely identifying the USB controller involved.	Optional
name	Specifies the user-friendly name of the device.	Optional
productID	Specifies the product ID of the device.	Always
type	Specifies the type of the device, where <code>generic</code> means the device is attached to the local ESXi host and <code>remote</code> means the device may reside on another host (or a client machine) and has been attached to the virtual machine through a network connection. Other values are specific to the virtual device ( <code>hid</code> and <code>hub</code> for example).	Optional
vendorID	Specifies the vendor ID of the device.	Always

## vm.usb.disconnect

94

This event is generated when an attempt was made to disconnect a USB device from a virtual machine.

Parameter	Description	Presence
subject	The value is always an empty string (indicating activity performed by the system rather than a user).	Always
object	Specifies the path of the VMX file of the reporting virtual machine.	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
opID	Specifies the operation ID.	Optional
deviceID	Specifies the device ID of the device.	Optional
deviceLabel	The value has the form <code>usb:&lt;N&gt;</code> for usb 1.1, <code>ehci:&lt;N&gt;</code> for usb 2.0, or <code>usb_xhci:&lt;N&gt;</code> for usb 3.x. The <code>&lt;N&gt;</code> is a small integer, uniquely identifying the USB controller involved.	Optional
name	Specifies the user-friendly name of the device.	Optional
productID	Specifies the product ID of the device.	Always
type	Specifies the type of the device, where <code>generic</code> means the device is attached to the local ESXi host and <code>remote</code> means the device may reside on another host (or a client machine) and has been attached to the virtual machine through a network connection. Other values are specific to the virtual devices ( <code>hid</code> and <code>hub</code> for example).	Optional
vendorID	Specifies the vendor ID of the device.	Always

## x509.cacert.add

95

This event is generated when an attempt was made to add a CA certificate to the host CA store.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always an empty string (representing the host itself).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional
subjectDN	Specifies the subject distinguished name of the CA certificate.	Always



## x509.cacert.remove

96

This event is generated when an attempt was made to remove a CA certificate from the host CA store.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always an empty string (representing the host itself).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional
subjectDN	Specifies the subject distinguished name of the CA certificate.	Always

## x509.srvcert.generate

97

This event is generated when an attempt was made to generate a client certificate signing request.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always an empty string (representing the host itself).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional

## x509.srvcert.install

98

This event is generated when an attempt was made to install a new server certificate.

Parameter	Description	Presence
subject	Specifies the name of the user who requested the operation.	Always
object	The value is always an empty string (representing the host itself).	Always
result	The value is either <code>success</code> or <code>failure</code> .	Always
reason	Provides additional details about the event.	Optional
ip	Specifies the network address of the originator of the operation.	Optional
opID	Specifies the operation ID.	Optional
subjectDN	Specifies the subject distinguished name of the server certificate.	Optional