bytes_written_count

registry

The bytes_written_count field provides a numeric count of how many bytes were written to the respective registry location.

EQL Query Example:

Show registry events where the number of bytes written is equal to 4:

```
registry where bytes_written_count == 4
```

bytes_written_string

registry

The bytes_written_string field provides a human readable value of the content that was written to the respective registry location.

EQL Query Example:

Show registry events where the string written contains "en-US":

```
registry where bytes_written_string == "*en-US*"
```

command_line

process

The command_line field contains the command line arguments passed through to the respective process.

EQL Query Example:

Show powershell.exe downloading files:

```
process where command_line == "*invoke-webrequest*" or command_line ==
"*downloadfile*"
```

destination_address

network

The destination_address field provides the destination IP address of the respective network activity.

EQL Query Example:

Show all network activity to the destination address 8.8.8.8:

```
network where destination_address == "8.8.8.8"
```

destination_port

network

The destination_port field provides the destination port address of the respective network activity.

EQL Query Example:

Show all network activity to the destination port 4444:

```
network where destination_port == 4444
```

event_id

security

The event_id field provides the Microsoft Event ID for the respective security event. Below is a breakdown of Windows Security Events collected by the Endgame sensor:

4672	Admin Logon
4648	Explicity User Logon
4647	User Logoff
4624	User Logon
4625	User Logon Failed
4801	Workstation Unlocked
4800	Workstation Locked

EQL Query Example:

Show all successful Windows logon events:

```
security where event_id == 4624
```

file_name

file

The file_name field provides the string name of the respective file..

EQL Query Example:

Find all events where the file_name is "badness.txt":

```
file where file_name == "badness.txt"
```

file_path

file

The file_path field provides the full path of the respective file.

EQL Query Example:

Show file events where the file_path contains "C:Windows\Prefetch":

```
file where file_name == "badness.txt"
```

image_name

image_load

The image_name field provides the string name of the respective image loaded by a process.

EQL Query Example:

Show all events where the image "vaultcli.dll" is loaded by a process:

```
image_load where image_name == "vaultcli.dll"
```

image_path

image_load

The image_path field provides the path of the respective image loaded by a process.

EQL Query Example:

Show all events where a process loaded an image from the path "C:\Windows\System32":

```
image_load where image_path == "C:\\Windows\\System32\\*"
```

total_in_bytes

network

The in_bytes field provides the total number of bytes received by the endpoint during the respective network activity.

EQL Query Example:

Show all network activity where the host received exactly 800 bytes:

```
network where total_in_bytes == 800
```

ip_address

security

The ip_address field provides the IP Address of the respective endpoint the security event took place on.

EQL Query Example:

Show all security events that occurred on 192.168.1.10:

```
security where ip_address == "192.168.1.10"
```

key_path

registry

The key_path field provides the path where the respective registry event occurred.

EQL Query Example:

Show all registry events with a key_path of "\Software\Microsoft\Windows\CurrentVersion\RunOnce*":*

```
registry where key_path ==
"*\Software\Microsoft\Windows\CurrentVersion\RunOnce"
```

key_type

registry

The key_type field provides the data type of the respective registry key. Below is a list of available registry data types:

binary	Binary data.
dword	32-bit number.
multiSz	Array of null-terminated strings that are terminated by two null characters.
qword	64-bit number.
sz	Null terminated Unicode or ANSI string.
unknown	No defined registry data type.

EQL Query Example:

Show all sz data type registry events:

```
registry where key_type == "sz"
```

logon_type

security

The logon_type field provides information regarding how the respective user logged on to the target endpoint. A list of logon types is provided below:

2	Interactive (logon via keyboard/screen of endpoint).
3	Network (connection to shared folder on endpoint from elsewhere on the network).
4	Batch (i.e. Scheduled Task).
5	Service
7	Unlock
8	NetworkCleartext (Logon with credentials sent via clear text – i.e. IIS logon with basic authentication).
9	NewCredentials (RunAs or mapping network drive with different set of credentials).
10	RemoteInteractive (Terminal Services, RDP, Remote Assistance).
11	CachedInteractive (Logon via cached credentials).

EQL Query Example:

Show all interactive logon events:

```
security where logon_type == 2
```

md5

process image_load

The md5 field provides a 128-bit hash value of the respective process or image loaded by a process.

EQL Query Example:

Find all processes with the MD5 hash of a37ed7663073319d02f2513575a22995f:

```
process where md5 == "a37ed7663073319d02f2513575a22995"
```

old_file_name

file

The field old_file_name provides the original file name of the respective file.

EQL Query Example:

Find all files with the old_file_name of "badness.tmp":

```
file where old_file_name == "badness.tmp"
```

old_file_path

file

The field old_file_path provides the original file path of the respective file when it is renamed or moved.

EQL Query Example:

Find all files with the original_file_path of "C:\Windows\Temp":

```
file where original_file_path == "C:\\Windows\\Temp\\*"
```

out_bytes

network

The out_bytes field provides the total number of bytes transmitted by the endpoint during the respective network activity.

EQL Query Example:

Show all network activity where the host sent exactly 800 bytes:

```
network where out_bytes == 800
```

parent_process_name

process

The parent_process_name field provides the parent process responsible for invoking the respective target process.

EQL Query Example:

Find the child processes of the parent process "wscript.exe":

```
process where parent_process_name == "wscript.exe"
```

parent_process_path

process

The parent_process_name field provides the parent process responsible for invoking the respective target process.

EQL Query Example:

Find parent processes located in the parent_process_path of "C:\Windows\System32":

```
where parent_process_path == "C:\\Windows\\System32\\*.exe"
```

pid

dns registry process network file security image_load

The pid field provides the process identifier (PID) of the respective target process. Please note that PIDs will be recycled by the operating system.

EQL Query Example:

Find a process assigned the pid of 4:

```
process where pid == 4
```

process_name

dns registry process network file security image_load

The process_name field provides the string name of the current target process.

EQL Query Example:

Find processes with a process_name of "mimikatz.exe":

```
process where process_name == "mimikatz.exe"
```

process_path

dns registry process network file security image_load

The process_name field provides the path of the current target process.

EQL Query Example:

Find processes with a process_path of "C:\Users\AppData":*

```
process where process_path == "C:\\Users\\*\\AppData\\*"
```

protocol

network

The protocol field provides the network protocol utilized during the respective network activity. Below is a list of network protocols:

tcp	Transmission Control Protocol
udp	User Datagram Protocol
unknown	Assigned when a network protocol is unidentified./

EQL Query Example:

Find processes with a process_path of "C:\Users\AppData":*

```
process where process_path == "C:\\Users\\*\\AppData\\*"
```

query_name

dns

The query_name field provides the string of the DNS resource the endpoint attempted to access.

EQL Query Example:

Find processes all queries to www.google.com:

```
dns where query_name == “www.google.com”
```

sha1

process image_load

The sha1 field provides 160-bit hash value of the respective process.

EQL Query Example:

Find all process events with a sha1 hash of “005754dab657ddc6dae28eee313ca2cc6a0c375c”:

```
process where sha1 == “005754dab657ddc6dae28eee313ca2cc6a0c375c”
```

sha256

process image_load

The sha1 field provides 160-bit hash value of the respective process.

EQL Query Example:

Find all process events with a sha256 hash of “a78c9871da09fab21aec9b88a4e880f81ecb1ed0fa941f31cc2f041067e8e972”.

```
process where sha1 ==
“a78c9871da09fab21aec9b88a4e880f81ecb1ed0fa941f31cc2f041067e8e972”
```

signature_signer

process image_load

The signature_signer field provides the entity responsible for code signing the respective process.

EQL Query Example:

Find all processes signed by Microsoft:

```
process where signature_signer == “*Microsoft*”
```

signature_status

process image_load

The signature_status field provides the current signature validity of the respective process. Below are the possible signature_status states:

trusted	The signature status is trusted.
untrusted	The signature status is not trusted.
noSignature	No signature is found.
errorUntrustedRoot	Trusted Root Certificate does not exist in Certificate Store
errorBadDigest	Certificate hash does not match

EQL Query Example:

Find all processes with a valid certificate:

```
process where signature_status == "trusted"
```

source_address

network

The source_address field provides the source IP address of the respective network activity.

EQL Query Example:

Show all network activity from the source address 8.8.8.8.

```
network where source_address == "8.8.8.8"
```

source_port

network

The source_port field provides the source port of the respective network activity.

EQL Query Example:

Show all network activity where the source port is 4444.

```
network where source_port == 4444
```

timestamp_utc

dns registry process network file security image_load

The timestamp_utc field provides a Coordinated Universal Time when the respective event occurred.

EQL Query Example:

Find all DNS events that occurred at 2019-02-07 18:46:00Z:

```
security where timestamp_utc == "2019-02-07 18:46:00Z"
```

unique_pid

The unique_pid field provides a unique process identifier for each process that is assigned by the Endgame sensor. This field differs from an operating system assigned process identifier (PID) as it avoids PID reuse.

EQL Query Example:

Show all process events with the unique_pid of 33458:

```
process where unique_pid == 33458
```

user_name

The user_name field provides the string name of the respective user.

EQL Query Example:

Show all process events of the user "arnold":

```
process where user_name == "arnold"
```