Splunk® Enterprise Security
Administer Splunk Enterprise Security 7.0.1

Create sequence templates in Splunk Enterprise Security

Generated: 5/03/2022 12:38 pm

Create sequence templates in Splunk Enterprise Security

The Event Sequencing Engine provides capabilities for threat detection that allow you to group correlation searches into batches of events, either in a specific sequence, by specific attributes, or both.

You create batches of events by defining a workflow to run correlation searches in an order of your choice, specifying what notable events would need to occur in order to advance to the next step.

The concept is similar to writing a script to automate the things that you might otherwise have to do manually when tracking a variety of notable events and variables through a variety of correlation searches. The concept is also similar to that of meta notable events or named multi-vector notables, which are alerts that are generated by correlation searches monitoring for multiple specific conditions prior to raising the alert.

How sequence templates work

The Event Sequencing Engine runs as a real-time search and listens for incoming notable events and risk modifiers that are triggered by correlation searches. Security analysts can provide specifications on how sequenced events are constructed by using sequence templates. Once you have created a sequence template, it is available for execution within 5 minutes.

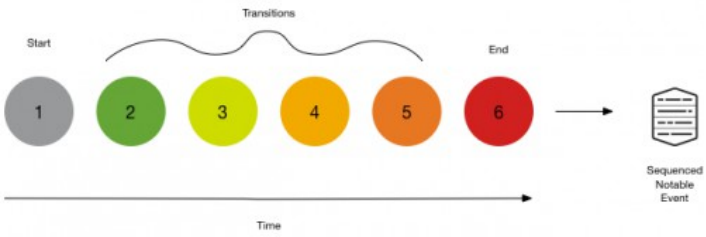
Sequence templates are stored in the `sequence_templates.conf` file.

The Event Sequencing Engine periodically stores information regarding the currently running sequence templates. This information can be viewed from the sequence lister page. See the status of a template.

Sequence template details

A sequence template defines the various constraints of constructing a sequence. It has three main components: start, transitions, and end. You can construct a sequence template using the editor.

The following diagram shows an example of the way that you can start with one correlation search (1), flow through any number of correlation searches in the transitions (2 through 5), and end with a final search (6).



Start

The start section defines match conditions for starting the execution of a template. Optionally, the start section can define state variables to store field values for the purpose of matching further notables or risk modifiers. State variables can also be used as outputs in the final sequenced event. Once the start condition is met, the event sequencing engine will start the execution of the corresponding sequence template.

Match conditions

The match condition defines the criteria for considering notable events or risk modifiers for transitioning through the phases in a template. The match condition has two parts that are evaluated successively, correlation search and expression.

Match Conditions	Description
Correlation Search	The correlation search to match the source of the incoming notable or risk modifier. Wildcard matching (*) is supported on this field.

Expression	The expression allows you to compare any field from an incoming event with a static value or a state variable (see state for further information). Expressions follow Splunk style syntax in the format of `<field> <comparator> <value>`.
	
	Note that while similar to SPL syntax, expressions in the event sequencing engine are more restricted than in standard SPL syntax. For example, SPL doesn't enforce AND/OR operators for field searches, but the event sequencing engine does. Wildcard matching (*) and regular expressions are not supported in the expression section.
	
	You can also use brackets for grouping. You must use the logical operators of AND or OR in your grouping, such as: `'host' = "127.0.0.1" AND ('dest' = "example.com" OR 'dest'="example.org")`. The NOT operator is not supported.
	
	The expression is made up of field, comparison function, and value.
	
	Field
	The name of any SPL field in single quotes, such as: `'host'`, `'source'`, `'sourcetype'`, etc. Multivalue fields are supported, and an event is considered a match as long as one value matches.
	Comparison function
	The comparison function can be any of the following: `=`, `!=`, `>`, `<`, `>=`, `<=`.
	The following comparison functions force numeric comparisons: `>=`, `<=`, `>`, `<`.
	Value
	The field value in double quotes, which can be in string format or in state_variable notation, such as: `'host' = "127.0.0.1"` or `'host' = "$host$"`.
	
	Based on an example event in .csv format such as the following:
	
	`host, source, sourcetype`
	`127.0.0.1, "Threat Detected", "nginx"`
	
	• An expression for matching on the host is `'host' = "127.0.0.1"`.
	• An expression for matching on any other source is `'source' != "Threat Detected"`.
	
	If you want to use assets and identities in expressions, configure asset and identity correlation with the **enable for all sourcetypes** option selected. This makes sure that identity and asset information is enriched during search time when receiving contents from the risk or notable index. See Configure asset and identity correlation in Splunk Enterprise Security.

State

The state provides a way to store values from matched events for the lifetime of a sequence. State can be stored at the start section and at each transition if the enforce ordering check box is unchecked. You cannot save a new state at the end step. These values can then be used for matching expressions in consecutive transitions. State can also be an output in the final sequenced event. If a multivalue is the output in the final sequenced event, it will be returned in a comma

separated format. Once stored, state variables can be referenced using `$variable_name$` syntax. State allows you to store important pieces of information for future matching. The state contains two parts, the field and the label.

State	Description
Field	The name of any SPL field that you want to capture for later use. State fields defined in the start section can be used in all transitions. But state fields defined in the transitions section will be available only to expressions in subsequent transitions.
Label	The label is the variable name for referring to the state field in a later search. The state_variable notation for referring to the label is similar to an SPL token in the way it is used to capture and pass values. The label is available only for use while the template is running. It does not persist when the template terminates or completes. The label cannot contain a dollar sign ($).

Transitions

The transitions section defines the sequence, either chronologically or in an order-independent way. You can define a series of match conditions to find the sequence. Each transition defines a title and a match condition.

Chronological

Transitions are matched chronologically by default. With the **enforce order** check box checked, the Event Sequencing Engine will check if notable events or risk modifiers match the completed transitions in the order specified. A transition is completed by matching an incoming event with a match condition. Given a sequence of correlation searches in the following order, with the **enforce order** check box checked for example, the notable events will be matched in order:

Start
1. Brute Force Access Behavior Detected

Transitions
2. Uncommon Processes On Endpoint
3. Unusually Long Command Line
4. Suspicious Reg.exe Process
5. Web Uploads to Non-corporate Sites by Users

End
6. Abnormally High Number of Endpoint Changes By User

Transitions can only define state variables if the **enforce order** check box is checked. Enforcing the order provides a way to chronologically build a sequence. A state stored in an earlier transition is available for matching in later ones.

Not chronological

You can turn off chronological matching by unchecking the enforce order checkbox. With enforce order unchecked, the Event Sequencing Engine will check if notable events or risk modifiers match any of the incomplete transitions. Once matched, corresponding transitions will be considered complete. The order of events will not be considered. For example, given a sequence of correlation searches with the enforce order check box unchecked, you'll notice that notable events can match in any order:

Start

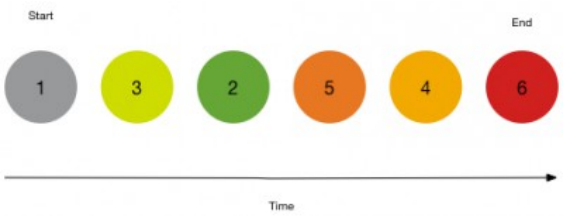
1. Brute Force Access Behavior Detected

Transitions

3. Unusually Long Command Line
2. Uncommon Processes On Endpoint
5. Web Uploads to Non-corporate Sites by Users
4. Suspicious Reg.exe Process

End

6. Abnormally High Number of Endpoint Changes By User



Wildcard

Transitions also support the same constructs for match conditions as in the start section. Since the correlation search field in the match condition allows wildcard match, it is possible to construct sequences that require forks. Transitions can define more than one next possible notable event or risk modifier. Given a wildcard correlation search sequence, for example:

The sequence can go in the following patterns:

Start

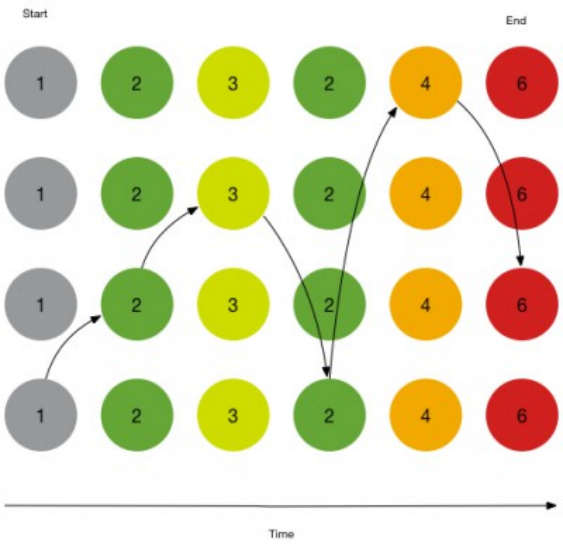
1. Brute Force Access Behavior Detected

Transitions

2. Option A or Option B, using a wildcard for two correlation searches. For example, a search of step_one or step_two by matching either one using the wildcard, such as `step*`.
4. Suspicious Reg.exe Process
5. Web Uploads to Non-corporate Sites by Users

End

6. Abnormally High Number of Endpoint Changes By User



Aggregate

Transitions can also be configured to aggregate notable events or risk modifiers that may happen after a transition match is found. If the **aggregate matches** check box is checked, the Event Sequencing Engine will add any notable events or risk modifiers that satisfy the match condition for one of the completed transitions. This can be used to add more context to the final sequenced event.

Consider a sequence of correlation searches like the following, where we have one correlation search that fires multiple notable events (Uncommon Processes On Endpoint) during the lifetime of our sequence:

Start
1. Brute Force Access Behavior Detected

Transitions
2. Uncommon Processes On Endpoint
3. Unusually Long Command Line
2. Uncommon Processes On Endpoint
4. Suspicious Reg.exe Process

End
6. Abnormally High Number of Endpoint Changes By User

If aggregate matches is unchecked, then there will only be one match for Uncommon Processes On Endpoint in the final sequenced event, even though it matched multiple times. If **aggregate matches** is checked, the event sequencing engine will try to match all new incoming notables and risk modifiers with completed transitions. In this case, after finding the first Uncommon Processes On Endpoint, the sequencing engine re-evaluates the next two Uncommon Processes On Endpoint notable events with match conditions and adds them to the final sequenced event if true.

End

The end section defines the termination criteria for a sequence template. A template can terminate if either of these two conditions are true:

- All transitions are complete and the event satisfying match condition is found. The event sequencing engine will consider this outcome as a successful run of a template and will trigger the sequenced event creation.
- The template has reached the configured max time to live (max_ttl). As the template has not reached its end state in the desired time, the event sequencing engine will discard this run and no sequenced event will be created.

Sequenced event

After the successful termination of a template, the output is a sequenced event. This sequenced event is the result of a template run and holds the necessary information for identifying a sequence. Sequenced events are written to the `sequenced_events` index. Sequence templates can be configured to use any of the state variables or statically configured values as output in the final sequenced events. The variables are stored and available for use only during the runtime of a template.

Create a template

You can create a template to run any number of searches that match your criteria.

The sequence template does not require any special capability to view, but requires the `edit_sequence_template` capability to manage sequence templates. By default, ES assigns the `edit_sequence_template` capability to the `ess_admin` role. An admin can assign it to other roles from the Permissions setting.

In the following scenario, you know that you're interested in detecting a prohibited application spawning the cmd.exe process. Once you've detected the process, you're interested in knowing if it's happening on your favorite computer, particularly if it starts creating new local admin accounts. Finally, you want to know if the user is making an abnormally high number of changes elsewhere. Because each system involved is set for logging at a different time interval, you are not necessarily interested in chronological order.

1. From the Splunk ES menu bar, select **Configure > Content > Content Management > Create New Content > Sequence Template**.
2. In the Sequence Template section, type a **Name** for your template, a **Description** for it, and select an **App** with which to run the search. If your template name has spaces, replace them with underscores.
3. In the Start section, add the following:
    1. Select the Correlation Search to begin with, such as **Detect Prohibited Applications Spawning cmd.exe**.
    2. Type the Expression to match on, such as `'dest' = "198.18.0.101"`
    3. Type a State to store for use in a later correlation search, such as:
        - **Field:** `user`
        - **Label:** `questionable_user`
4. In the Transition section, do the following:
    1. Uncheck the Enforce Ordering check box.
    2. Type a Title for this section, such as: `new local admin`
    3. Select the Correlation Search to run next, such as: **ESCU - Detect New Local Admin account - Rule**.
    4. Type the Expression to match on, such as the state you saved earlier: `'user' = "$questionable_user$"`.
5. In the End section, add the following:
    1. Select the Correlation Search to end with, such as **Change - Abnormally High Number of Endpoint Changes By User - Rule**.

    2. Type the Expression to match on, such as the state you saved earlier: `'user' = "$questionable_user$"`.
    3. Select the Time Limit when the search should expire, such as `2 days`.
6. In the Actions section, add the following:
    1. Type the Event Title that you want to see in the Incident Review, such as `Prohibited cmd, new local account, high endpoint changes`.
    2. Type the Description that you want to see in the Incident Review, such as `The questionable user on my favorite computer is $questionable_user$`.
    3. Select the Urgency that you want to see in the Incident Review, such as **High**.
    4. Select a Security Domain that you want to see in the Incident Review, such as **Access**.
7. Click **Save**.

Enable or Disable a template

Manage sequence templates individually by enabling or disabling each one. Enable or disable the template by performing the following steps:

1. From the Splunk ES menu bar, select **Configure > Content > Content Management**.
2. From the Type filter, select the **Sequence Template** option.
3. Check the check box for your Sequence Template.
4. Click **Edit selection > Enable** or **Edit selection > Disable**.

Disabling a sequence template automatically stops all the running instances of the template, as well as disabling future instances from starting. Reenabling a sequence template does not restart the instances that expired as a result of being disabled.

Enable event sequencing

Manage sequence templates as a whole by enabling or disabling the Event Event Sequencing Engine. The sequence templates will run only if the Event Sequencing Engine is enabled. The Event Sequencing Engine is disabled by default.

Enable the Event Sequencing Engine by performing the following steps:

1. From the Splunk ES menu bar, select **Configure > General > General Settings**.
2. (Optional) Type `Event Sequencing Engine` in the filter field.
3. Click **Enable** to enable the Event Sequencing Engine.

Edit an existing template

The sequence template does not require any special capability to view, but requires `edit_sequence_template` to manage sequence templates. By default, ES assigns the `edit_sequence_template` capability to the `ess_admin` role. An admin can always assign it to other roles from the Permissions setting.

You can edit all templates, whether they're enabled or disabled.

1. From the Splunk ES menu bar, select **Configure > Content > Content Management**.
2. From the Type filter, select the **Sequence Template** option.
3. (Optional) Click **Disable** to disable an enabled template.
4. Click the name of the search to edit the template parameters.

See the status of a template

You can see which sequences are running or completed.

1. From the Splunk ES menu bar, select **Security Intelligence** > **Sequence Analysis**.
2. From the Showing filter, select the **Running Templates** or **Completed Templates**.
3. From the event information column, click the greater than (**>**) symbol to expand the display.

You can see which templates are running and their current status in terms of which events have been matched and how many transitions have been completed.

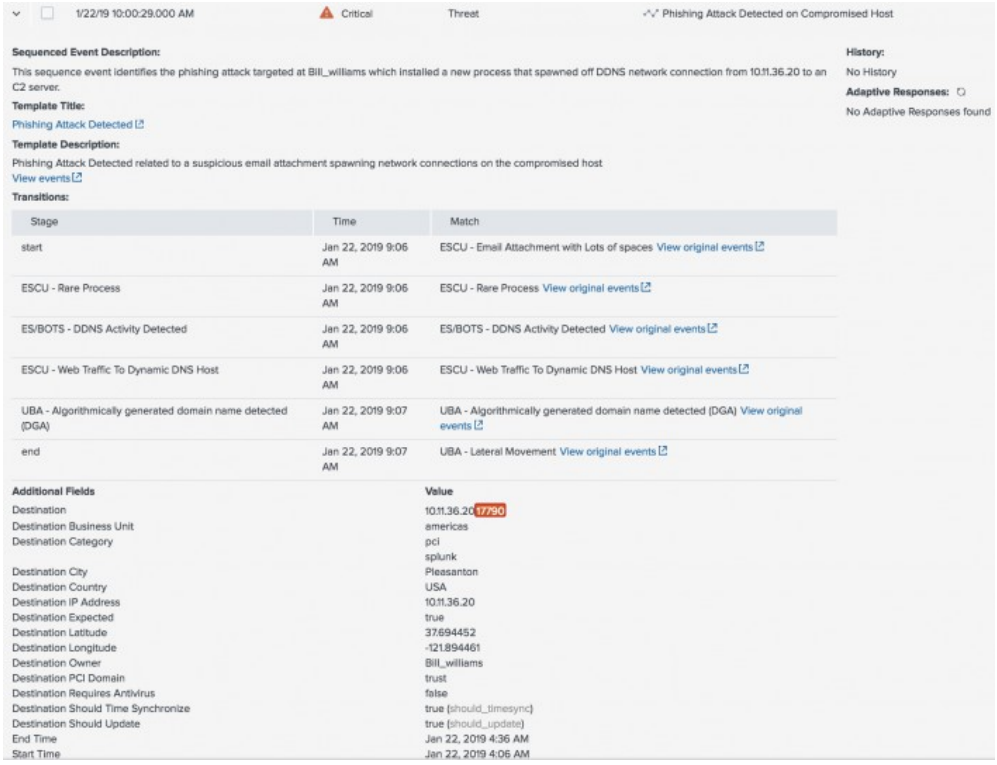
Find the sequenced events generated by the event sequence template

Once you create a sequence template and it reaches the end state, the output displays as a sequenced event in the Incident Review dashboard. See Incident Review overview for information about using the dashboard.

To find the output from the sequence template search, do the following:

1. From the Splunk ES menu bar, select **Incident Review**.
2. Click the **Sequenced Event** filter to show only sequenced events.
3. (Optional) Sort by **Title**.
4. You will see the Event Title that you typed in the editor as the title of your sequenced event.
5. From the event information column, click the greater than (**>**) symbol to expand the display.

ES displays information specific to that sequence of events, such as the name and description, the state of each transition in the sequence, and the sequence expiration date. For example when we see Rare Process, then DDNS Activity, then Web Traffic, then a UBA-triggered DGA alert.



Execute the Event Sequencing Engine in an ad-hoc manner

When you create a template, the Event Sequencing Engine starts executing it within 5 minutes. Alternately, you can run the helper macro, `execute_sequence_template`. This macro takes two parameters: the template name and a Boolean expression indicating if a sequenced event is created or not. For example:

```
`execute_sequence_template(template_name, false)`
```

In this case, `false` means that the sequenced event will not be created.

This macro can be run over historical data, so you can find sequenced events in past notable events and risk modifiers. After running the macro, the Event Sequencing Engine returns sequenced events if any are found. You can only execute one template at a time. This macro is intended for explorations and fine tuning to manage sequence templates.