Sentinel 7: Correlations for the Real World
Lecture

NIQ02 **Novell Training Services** www.novell.com

ATT LIVE 2012 LAS VEGAS

Novell®

Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the Novell International Trade Services Web page (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc., has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the Novell Legal Patents Web page (http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.

404 Wyman Street, Suite 500

Waltham, MA 02451

U.S.A.

www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the Novell Documentation Web page (http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Sentinel 7 Correlations for the Real World

Brad Toney
ATT Engineer
btoney@novell.com

Objectives

- Introduction
- Creating Correlations
- Case Study

Introduction

Sentinel Correlation

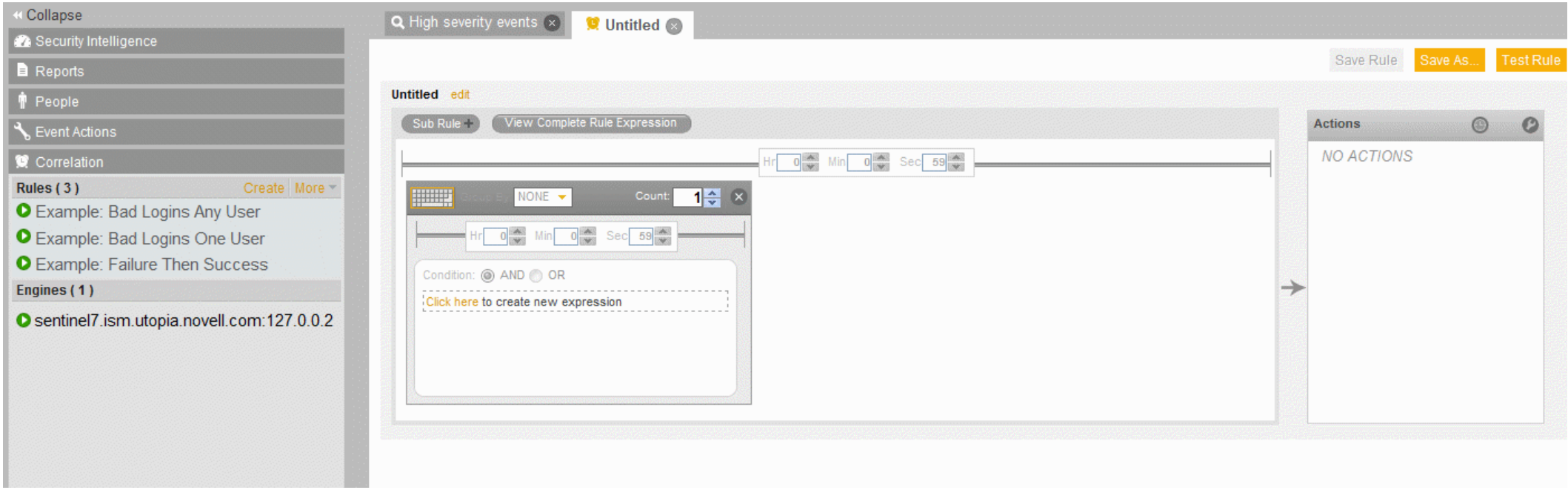
- Correlation rules define a pattern of events that should trigger, or fire, a rule.
- Sentinel helps you correlate such events with the rules you create and deploy in the Correlation engine and take appropriate action to mitigate any alarming situation.
- Create correlations using the correlation rule wizard or the simple RuleLG language, you can create rules that range from simple to extremely complex.

Administration

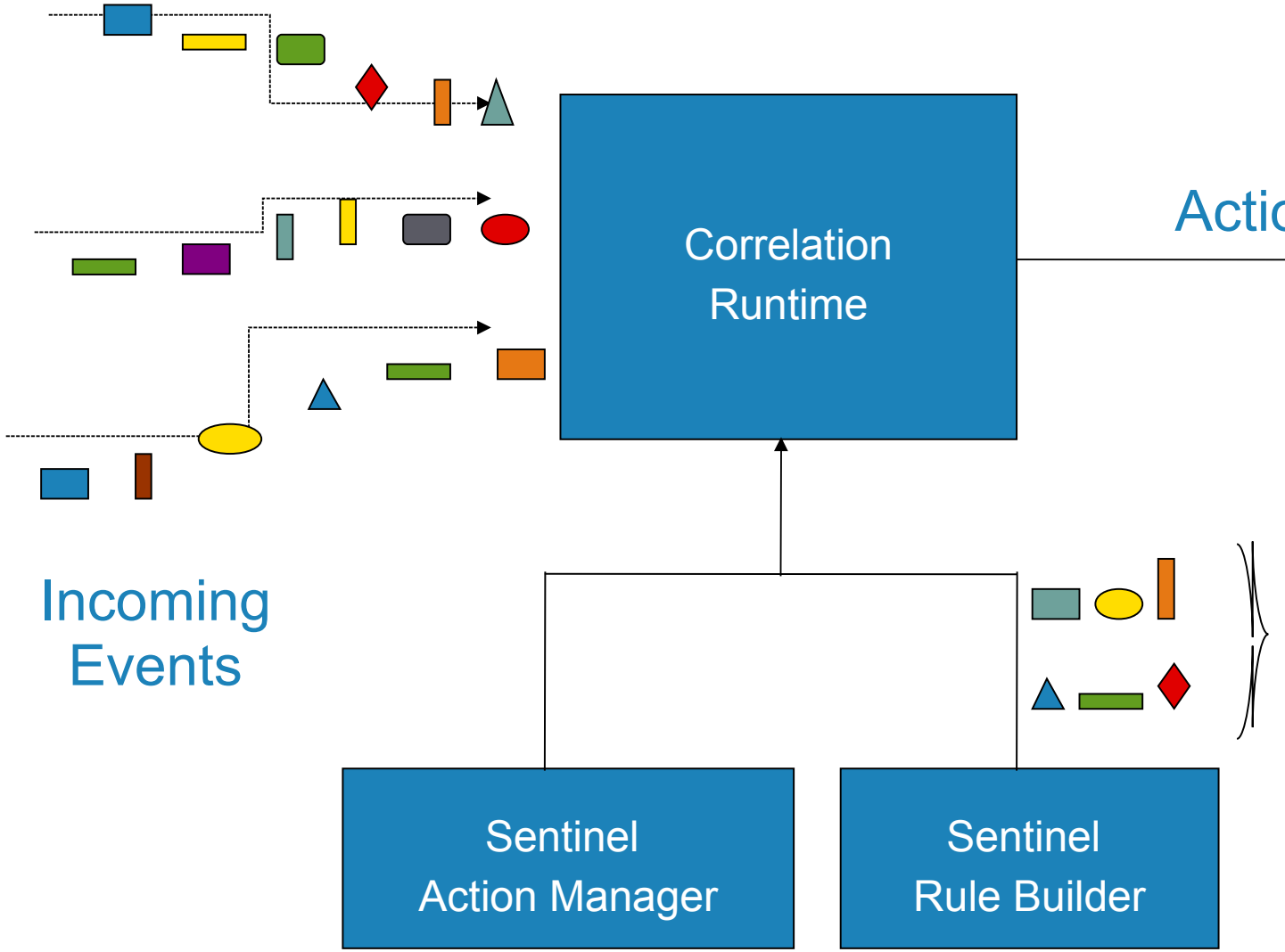
- Low Administration overhead
  - Centrally administer correlation engines
    - Start/Stop Engine
    - Enable/Disable Rules
    - Deploy rules
  - Graphically monitor status and activity information published by engines and rules
    - Rule Status
    - Number of events processed by engine/rule
    - Number of times rule fired

Creating Correlations

Correlations in the Web UI



Sentinel Correlation



Rule Language

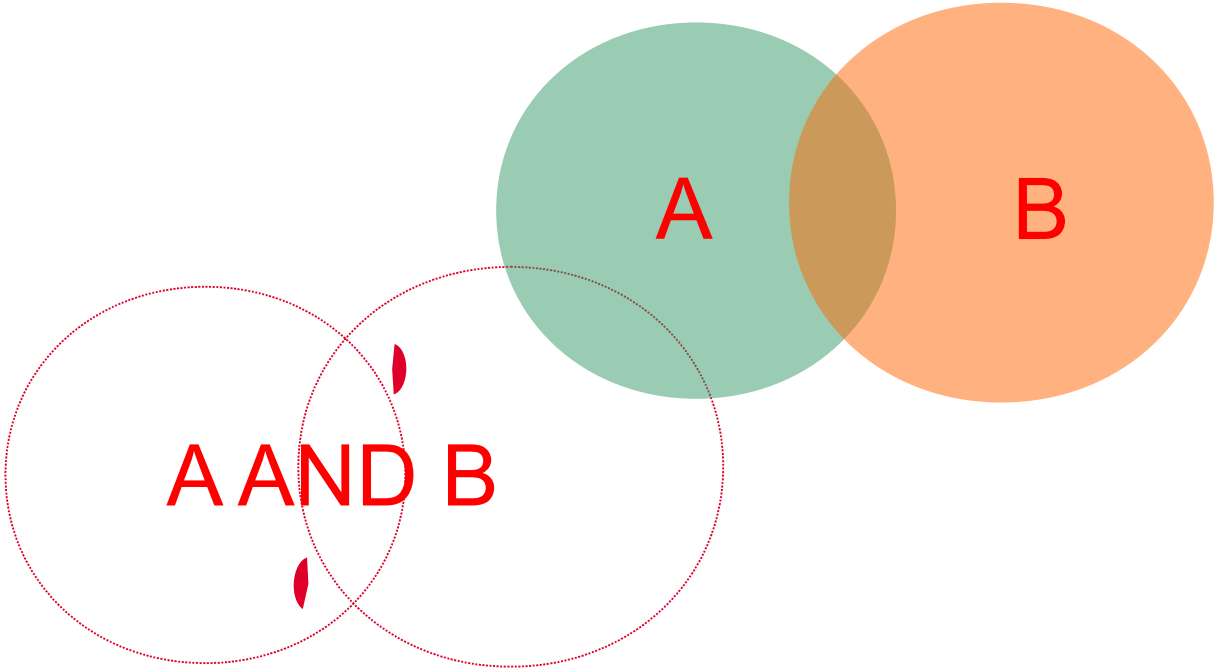
Constructs:

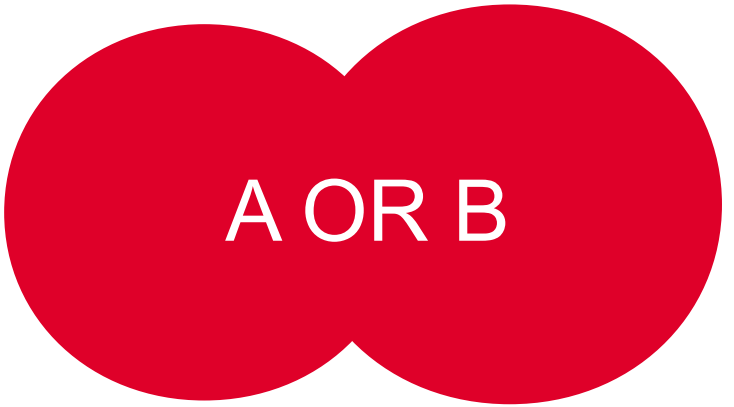
Filter
Window
Composite (Gate)
Sequence
Trigger (+ discriminator)

Operators:

flow
union
intersection

RuleLG Operators





A intersection B

A union B

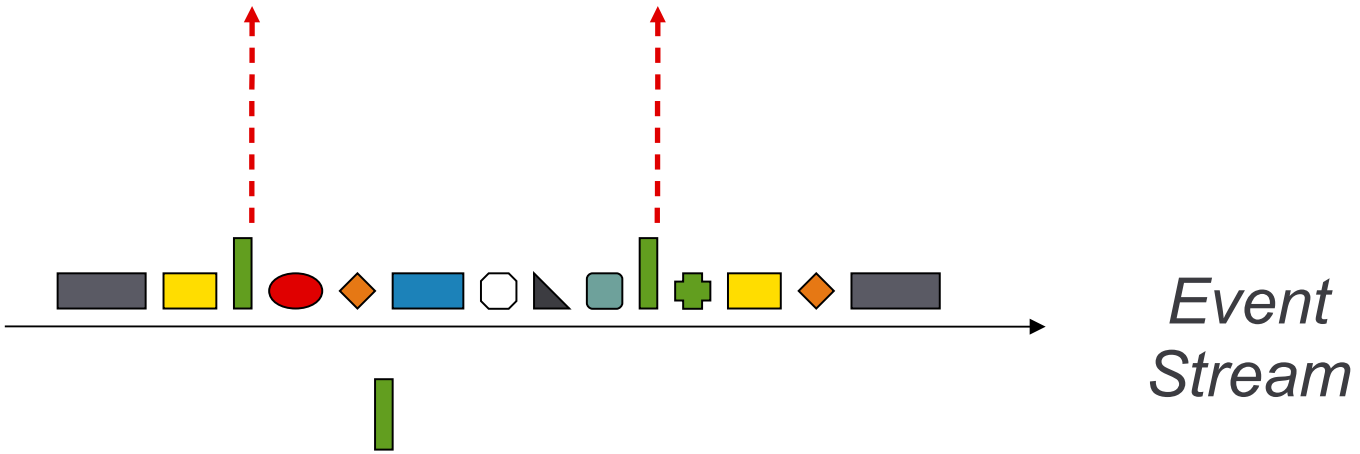
```
filter()  flow   window()   flow   trigger()

filter()  union  window()   union  trigger()
```

```
filter()  intersection  window()  intersection  trigger()
```

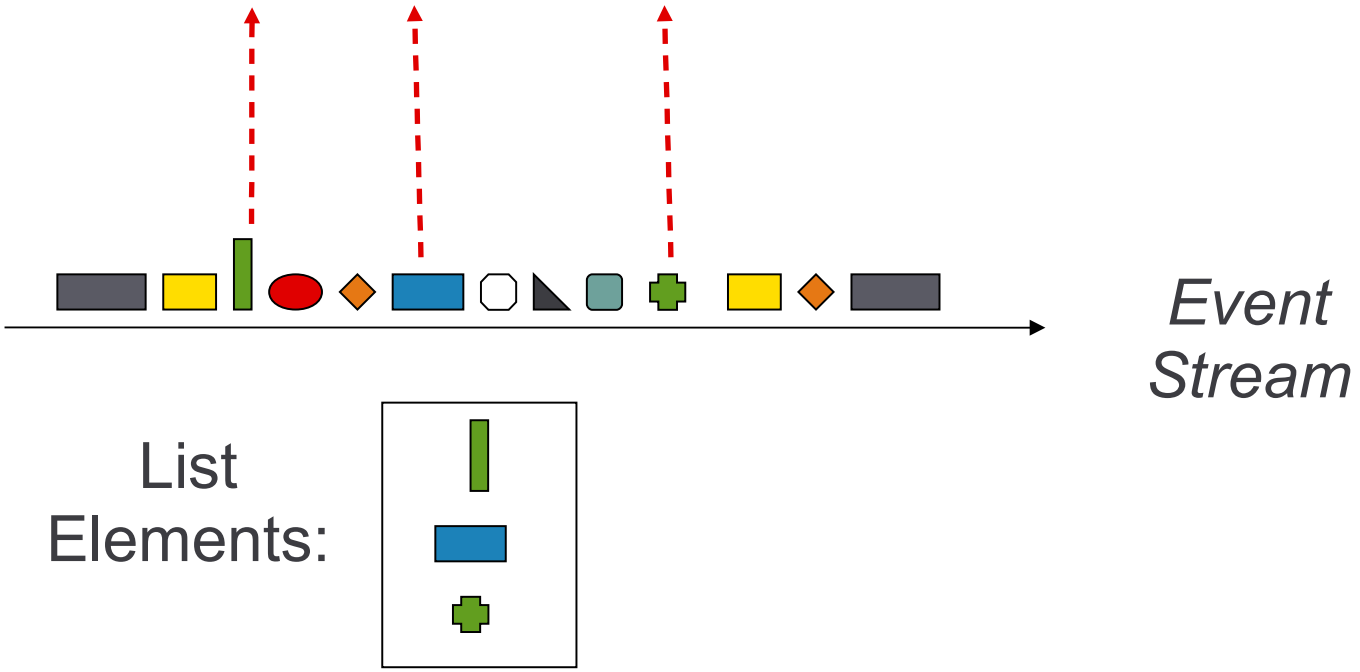
Filter Static Lookup

- Find an event with specific attributes
  - Severity 5 event from the Accounts server
    - filter(e.sev=5 and e.shn = “Accounts Server”)



Filter inList

- Severity 5 event from any critical server
  - filter(e.sev=5 and e.shn inlist CriticalServerList)



Filter inList

  - Unauthorized SMTP
    - filter(e.st = “F” and not (e.sip inlist AuthorizedMailServerList) and e.dp=25)

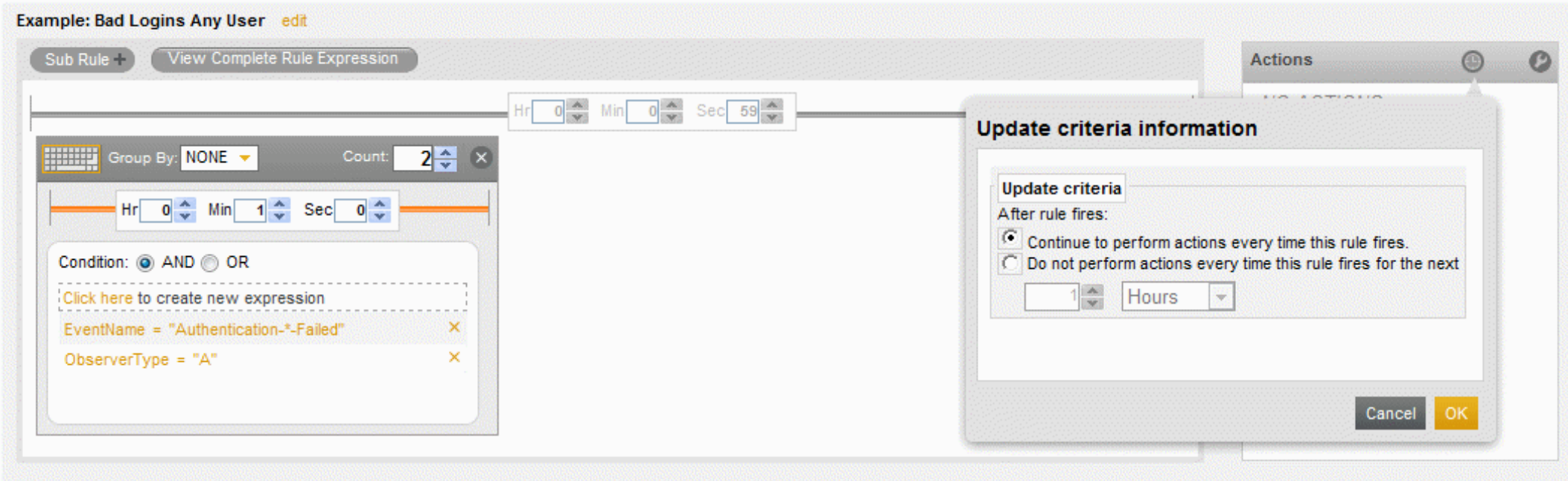
- “inlist” operator used to check data within existing dynamic lists

Filter isNull

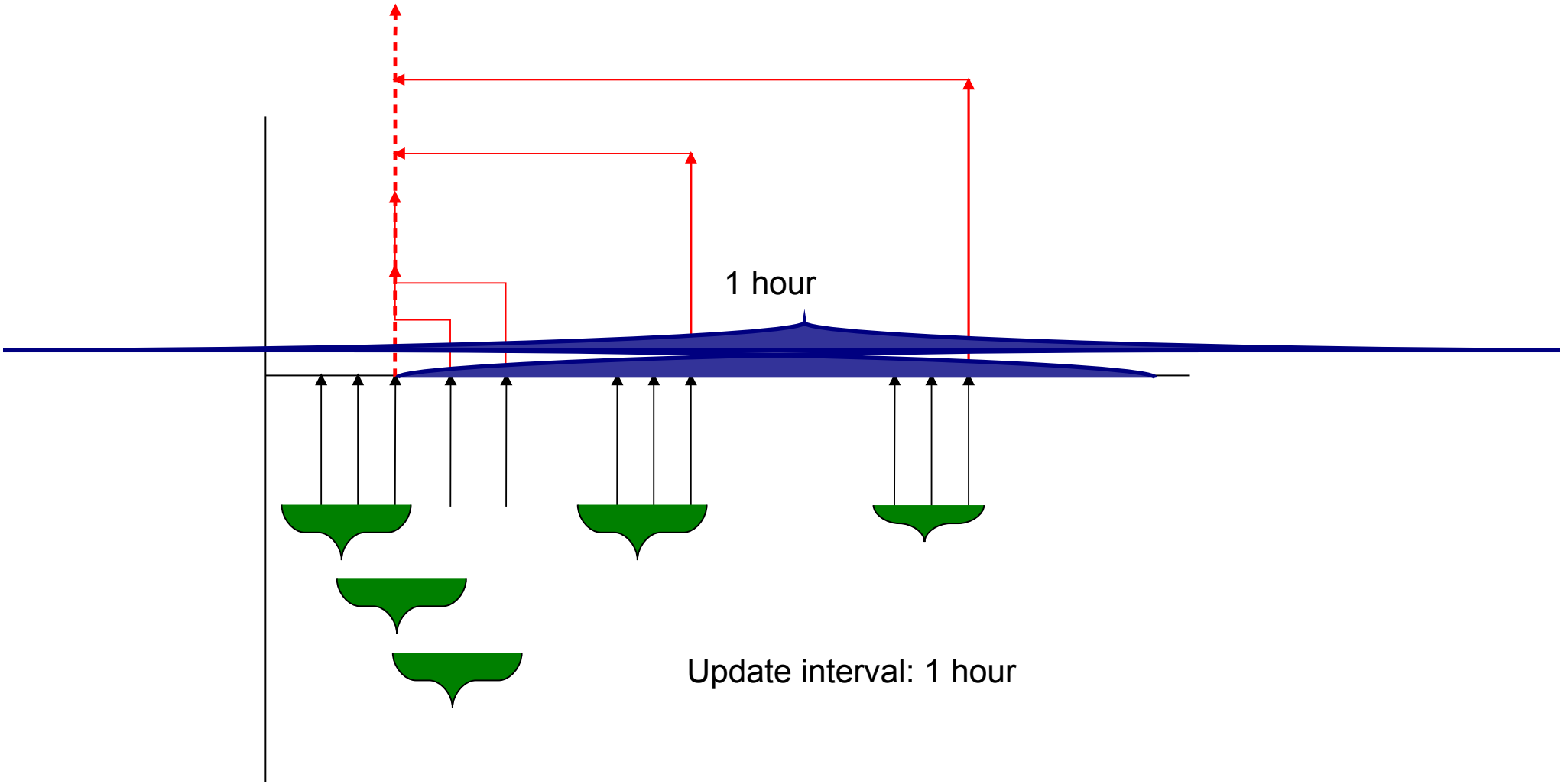
- Identify events with null attributes
- Always used within a filter
- Syntax
  - Filter(isNull(e.cv52) and e.evt=“file access”)
  - Filter(e.evt=“file access” and not(isNull(e.cv52)))

Correlation Updates Interval

- Allows us to treat recurring patterns as a continuation of the same attack.
- Time Based
- User Defines Update interval



Time interval



filter(e.evt = “failed login”) flow trigger(3, 30)

Basic Correlation

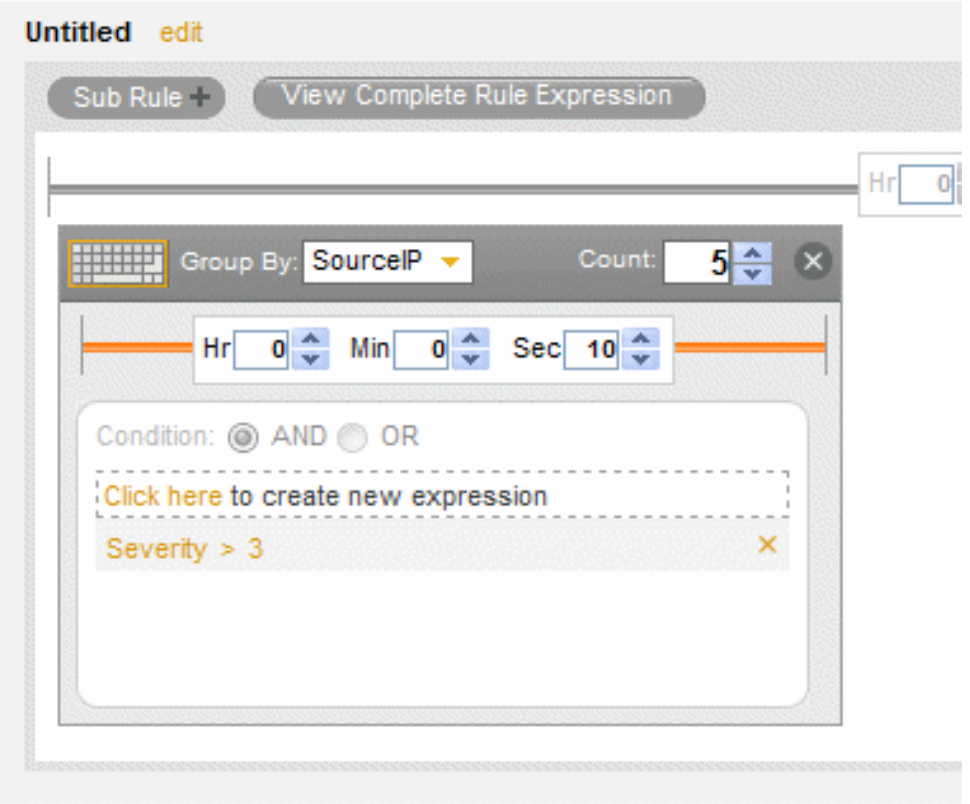
- We will start with basic functions and concepts (not meant to be used independently in a correlation)
- Next we will combine functions to make reasonable correlations
- Finally we will expand out Correlation vocabulary

RuleLG – Trigger Details

- **trigger(n,t)**
  - e.g. trigger(3, 90)
  - This operator will trigger when *n* meets the threshold of 3 keeping only *n-1=2* events in storage before triggering.
  - If *n=1*, then the operator will trigger immediately.
  - If *t=0*, the trigger is instantaneous; a special case used only for handling window output (see below)
- **trigger(n,t,discriminator(e))**
  - E.g. trigger(3,90,discriminator(e.sip))
  - This operator will trigger when *n* meets the threshold of 3 keeping *n-1*=2 events in storage for each distinct Source IP

RuleLG – Trigger Example

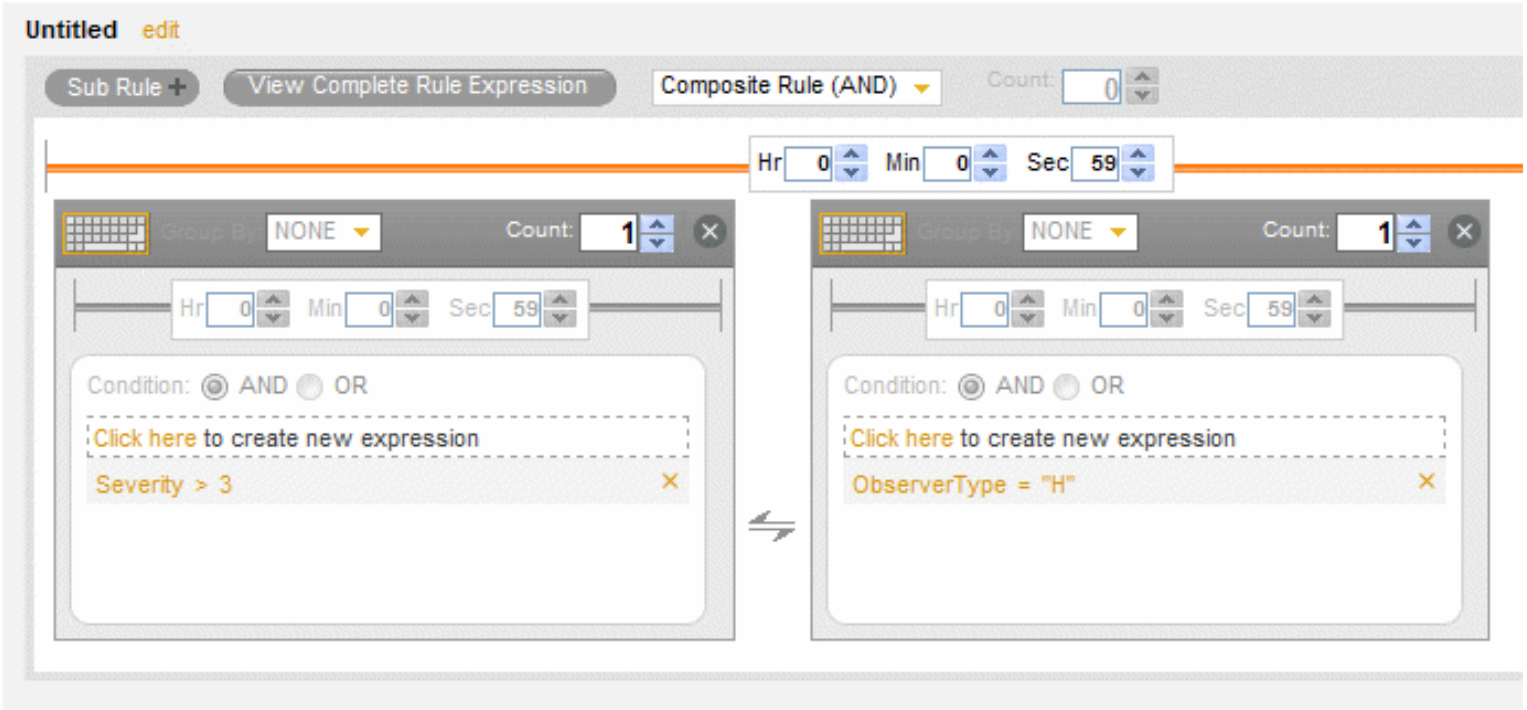
trigger(5,10, discriminator(e.SourceIP))
In this example, the trigger operation keeps distinct buckets for each unique Source IP; for each bucket, there is a count of 5 in 10 seconds.
If the threshold is met within the time frame, the trigger will generate a Correlated Event as an output set.



“Real-Life” Trigger RuleLG Example

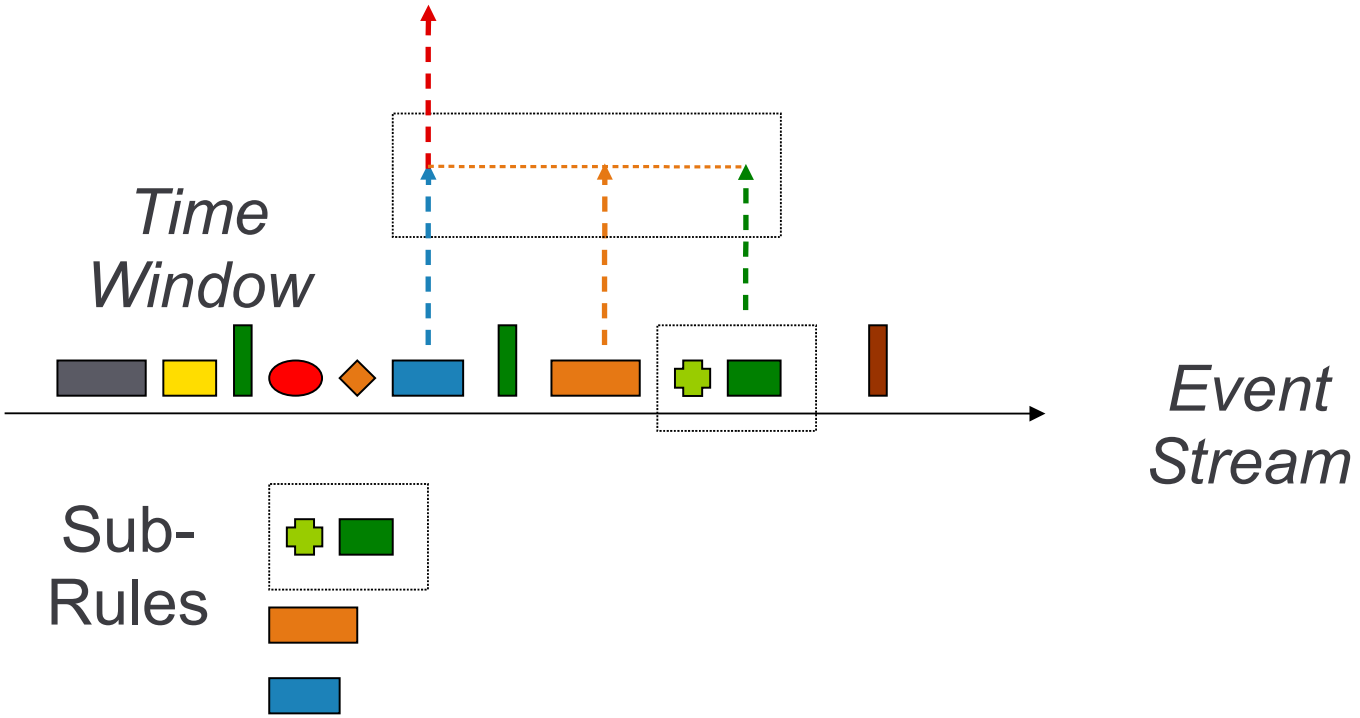
```
filter(e.DeviceCategory = "FW"
   and e.Severity >= 4) flow
trigger(5,600,discriminator(e.Des
tinationHostName))
```

Composite Rule



Gate

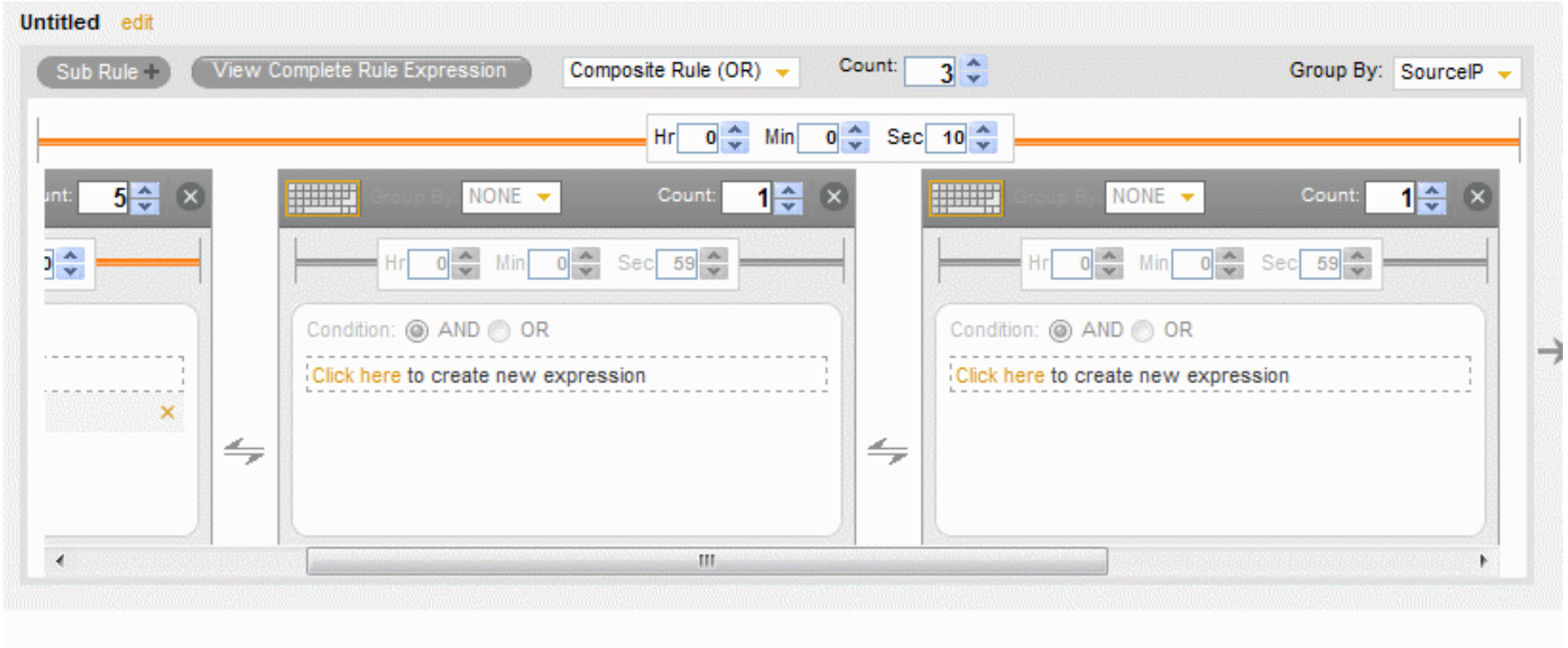
- Gate is composed of other rules
  - Sub-rule can be any rule, including another composite rule.
  - Rule fires if sub-rules fire within specified time period.
    - Configurable Mode: All, Any, Some (3 of 5)



Gate

- Syntax
  - gate(rule1, rule2, rule3...., <mode>, <time>, <discriminator>)
- Example
  - User logs in, changes password and withdraws more than 10,000 within a specified time frame.
    - gate(filter(e.evt="login"), filter(e.evt="passwordchange"), filter(e.evt="withdraw" and e.amt > 10000), all, 10m, discriminator(e.useraccount))

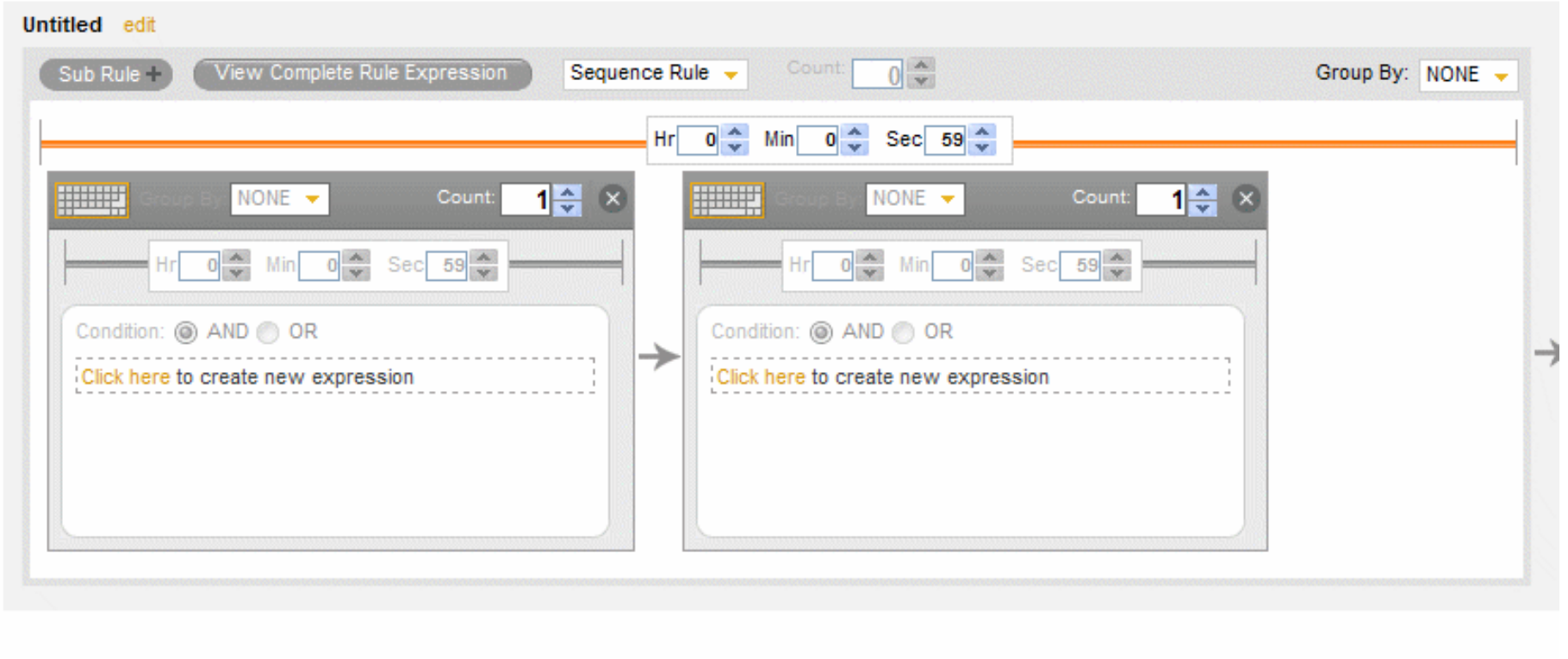
Composite Rule



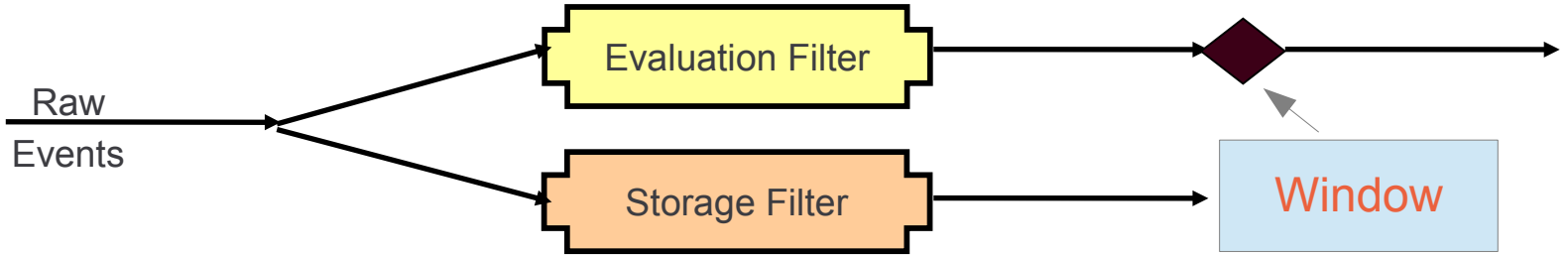
Sequence

- Complex pattern based on time-ordered sub-patterns.
- Rule fires only if all sub-rules fire in time order
- Syntax
  - sequence(rule1, rule2, rule3..., <time>, <discriminator>)
- Example
  - 3 failed logins followed by 1 successful login by the same user within 3 minutes.
    - sequence(filter(e.evt="login failed") flow trigger (3,180), filter(e.evt = "login"), 180, discriminator(e.sun))

Sequence



RuleLG – Window Filters



The window operator has two paths:

Evaluation path -

Input into the **window()** function, by default all events but can be pre-filtered using a standard **filter()** function.

Each incoming event is compared to events stored in the window by using the **Evaluation expression** (e.g.: w.sip = e.sip).

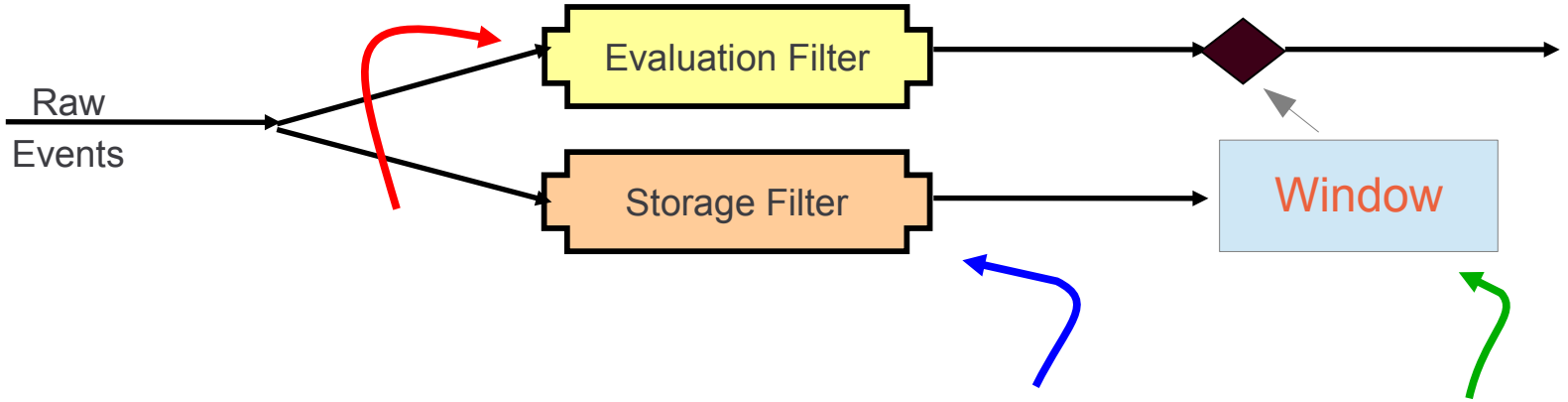
Storage path –

Determined by the **storage filter.**

A parallel stream of all raw events (but delayed by one event!) goes through the Storage Filter;

events that pass through the filter are added to the window.

RuleLG – Window Filters



The window operator has two paths:

Evaluation path `e.eventtag`

Stored bucket `w.eventtag`

Note that these two filters do **not** need to be the same. For example:

```
    filter(e.rv32="IDS") flow
window (w.dip = e.dip, filter(e.rv32="FW"),
        500) flow
    trigger(5,3600)
```

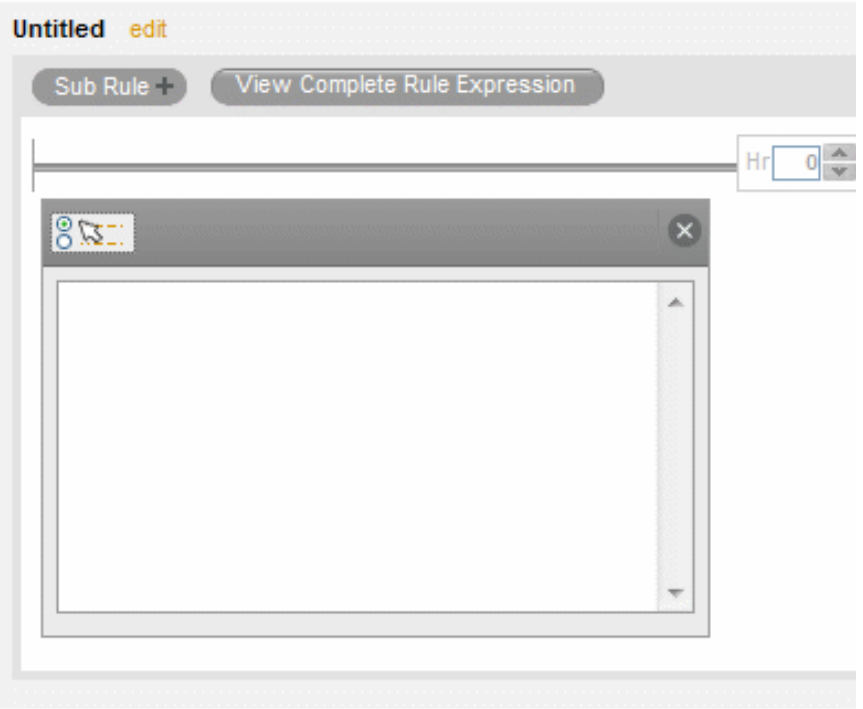
Major event from Firewall to the same Destination Host

```
filter(e.DeviceCategory = "FW"
    and e.Severity >= 4)
```

flow

```
trigger(5,600,discriminator
  (e.DestinationHostName))
```

Free Form RuleLG Features



RuleLG – Window

```
window(w.sip=e.sip, filter(e.sip
                match
 subnet(<192.168.12.0/24>)) ,60)
```

If the current event has a Source IP address that matches a past event's Source IP address that happened within the past 60 seconds, with the past events limited to those whose source IP is within the specified subnet, a correlated event is sent with the current event.

Window Example

Example:

```
filter(e.sev=2) flow window
(w.sip = e.sip, filter(e.sev=2),
500) flow trigger(5,3600)
```

“Real-Life” RuleLG Example

```
   filter(e.DeviceCategory = "FW" and
e.Severity >= 4)  flow window(w.SourceIP
 = e.SourceIP,filter(e.DeviceCategory =
  "FW" and e.Severity >= 4),120) flow
trigger(1,120,discriminator(e.SourceIP))
```

Major event from Firewall tracking SourceIP

filter(e.DeviceCategory = "FW" and e.Severity >= 4)

flow

window(w.SourceIP = e.SourceIP,filter(e.DeviceCategory = "FW" and e.Severity >= 4),120)

flow

trigger(1,120,discriminator(e.SourceIP))

Case Study

Case Study

This section presents several real world case studies.

They will involve:

A story describing what is happening

A control describing what the the company whats from Sentinel 7

An explanation about the security policy

The rule

And an observation about the rule.

Case Study 1 The Story

The Story:

In order to comply with regulations, the organization has to secure all the workstations to avoid access to classified information. is section presents several real world case studies.

One of the controls is to configure all terminals and sessions to block further authentication attempts, for a period of no less than 10 minutes, after 3 consecutive failed attempts from a user to login to his/her terminal.

Case Study 1 The Control

The Control:

The organization decides to use Sentinel 7 to monitor non compliant behavior through the use of correlations.

Specific to SSH in SUSEis section presents several real world case studies.

Case Study 1 The Explanation

The Explanation:

According to the organizations security policy, if all terminals are configured to block further authentication attempts after 3 failed attempts, is a fourth attempt is seen within 10 minutes if the first one, the workstation or terminal has not been configured correctly and is therefore non compliant.

Case Study 1 The Rule

The Rule:

A correlation rule that could be used to make sure that workstations and terminals are compliant might look like this.

filter(((e.EventName = “sshd: Authentication denied”)))flow trigger (4,600,discr4iminator(e,InitiatorUserName)

Case Study 1 An Observation

An Observation:

The previous rule applies specifically to SSH.

It would be more flexible to use taxonomy is possible.

Like the following example:

filter(e.XDASTaxonomyName=XDAS_AE_AUTHENTICATE_ACCOUNT AND (XDASOutcomeNAME=”XDAS_OUT_DENIAL” OR XDASOutcomeName=”XDAS_OUT_FAILURE”))flow trigger(4,600,descriminator(e.InitiatorUserName))

Case Study 2 The Story

The Story:

A system administrator at the bank is worried that external administrators from his service provider are abusing their privileged rights.

Upon investigation, he has found out that in order to hide their tracks, the external administrators would create a privileged user account with fake user details to avert suspicion.

Since all privileged user monitoring depends on filtering the KNOWN administrator accounts, the new account created would circumvent the controls because the username was not included in any of the monitoring lists.

Case Study 2 The Control

The Control:

To find out who else is creating fake privileged users and in order to monitor this illicit behavior, the security administrator uses Sentinel 7 to correlate the deviant behavior.

His investigation saw that the administrator would create the user and then use another account to delete it.

Case Study 2 The Rule

The Rule:

Based on the control, the security administrator create the following rule.

sequence(filter(((e.XDASTaxonomyName = “XDAS_AE_CREATE_ACCOUNT”))),filter(((e.XDASTaxonomyName = “XDAS_AE_DELETE_ACCOUNT”))) , 3600,discriminator(e.TargetUserName))

Case Study 2 An Observation

An Observation:

There may be a reason to increase the evaluation time period to more than just one hour.

If that were the case, the correlation rule rule should be run on a separate correlation server.

Case Study 3 The Story

The Story:

The organization has a development environment where application are constantly being tested.

This environment has been segregated from the rest of the production network and is treated as an UNTRUSTED segment of the network.

The auditor has asked to monitor the communications between this network and the production subnet.

Case Study 3 The Control

The Control:

The administrator is using Sentinel 7 correlations to monitor non compliant behavior from the development network.

The development network is 172.16.0.0/24

Case Study 3 The Rule A

The Rule:

Repetitive illegal communication detected by the firewall, usually indicates a worm wanting to break out.

This is more of an early threat detector.

filter(e.DeviceCategory = “FW” and e.XDASTaxonomyName = “XDAS_AE_TERMINATE_PEER_ASSOC” and e.XDASOutcomeName = “XDAS_OUT_DENIAL” and ((e.sip match subent(172.16.0.0/24))))flow trigger(1,60,descriminator(e.sip,e.dip))

Case Study 3 The Rule B

The Rule:

This rule is configured to do no action for 3 hours.

Every correlation event will contain non compliant communication between IP pairs.

filter(e.DeviceCategory = “FW” and ((e.sip match subent(172.16.0.0/24))) and ((e.dip match subnet(192.168.0.0/16))))flow trigger(1,60,descriminator(e.sip,e.dip))

Case Study 3 The Rule C

The Rule:

This rule is meant to alert on the non compliant behavior of using production logs to test systems by filtering known ports belonging to auditing and logging applications and DB ports.

Also configured to not do anything for X amount of hours.

filter(e.DeviceCategory = “FW” and (e.dp=512 or e.dp=1512 or e.dp=289 or e.dp=1289 or e.sp=1524 or sp=e.3306) and ((e.sip match subent(192.168.0.0/16))) and ((e.dip match subnet(172.16.0.0/24))))flow trigger(1,60,descriminator(e.sip,e.dip))

Lab exercise 1, 2, and 3

Lab exercise 1 will have you use a built in correlation rule.

Lab exercise 2 will have you create a simple correlation rule.

Lab exercise 3 will have you build the correlation rules associated with these case studies.

Sentinel 7 Correlations for the Real World

Brad Toney
ATT Engineer
btoney@novell.com

Objectives

- Introduction
- Creating Correlations
- Case Study

Introduction

Sentinel Correlation

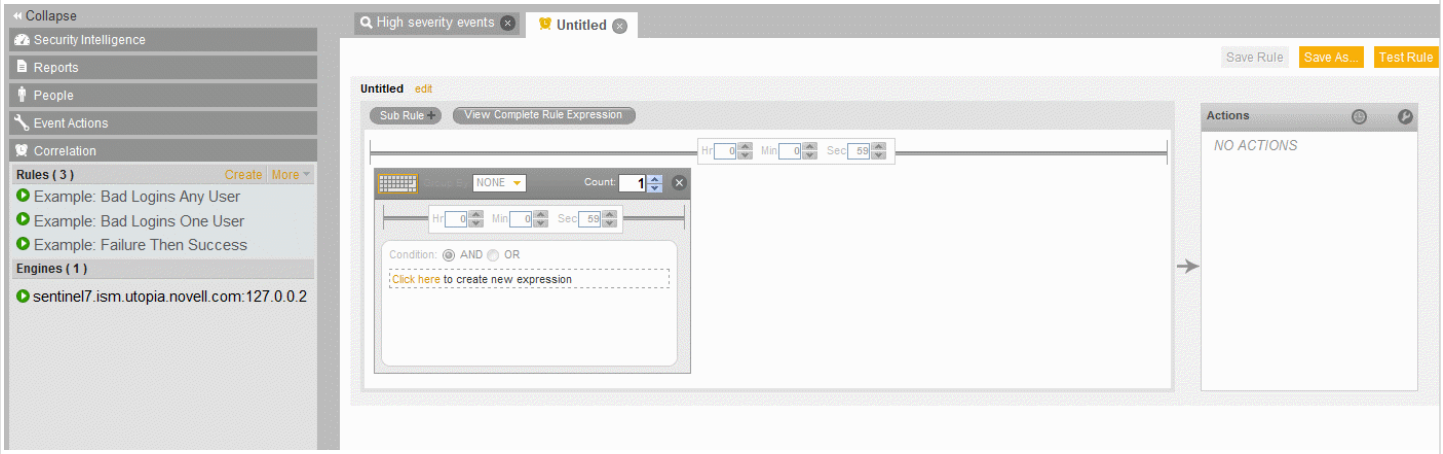
- Correlation rules define a pattern of events that should trigger, or fire, a rule.
- Sentinel helps you correlate such events with the rules you create and deploy in the Correlation engine and take appropriate action to mitigate any alarming situation.
- Create correlations using the correlation rule wizard or the simple RuleLG language, you can create rules that range from simple to extremely complex.

Administration

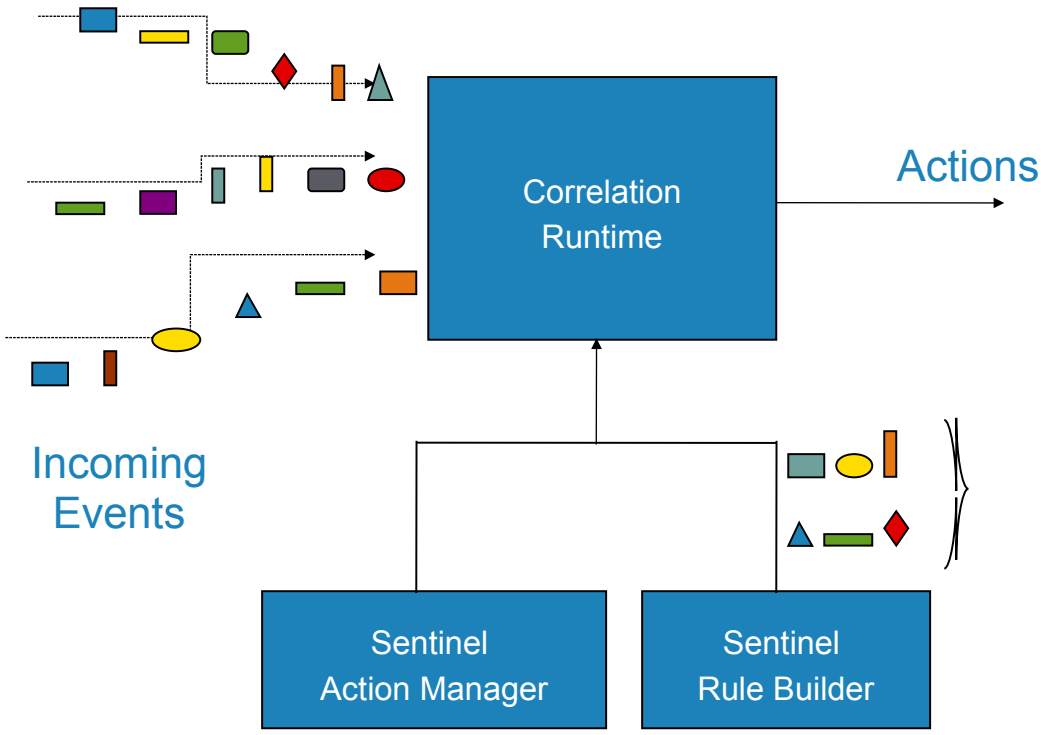
- Low Administration overhead
  - Centrally administer correlation engines
    - Start/Stop Engine
    - Enable/Disable Rules
    - Deploy rules
  - Graphically monitor status and activity information published by engines and rules
    - Rule Status
    - Number of events processed by engine/rule
    - Number of times rule fired

Creating Correlations

Correlations in the Web UI



Sentinel Correlation



Rule Language

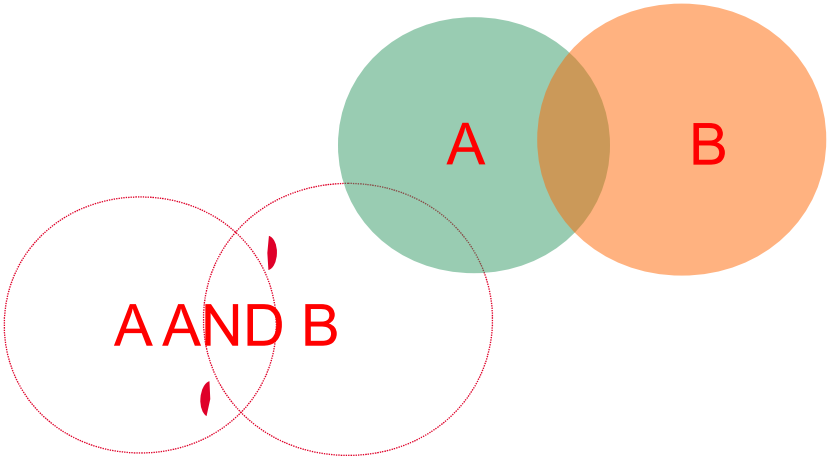
Constructs:

Filter
Window
Composite (Gate)
Sequence
Trigger (+ discriminator)

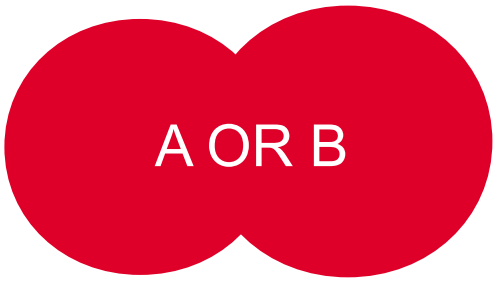
Operators:

flow
union
intersection

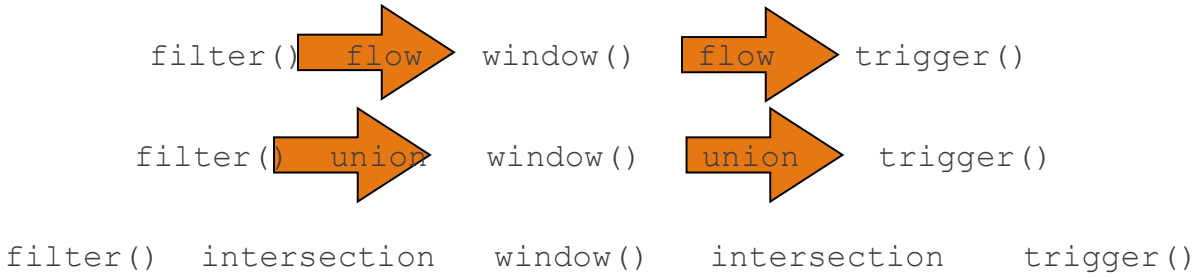
RuleLG Operators



A intersection B

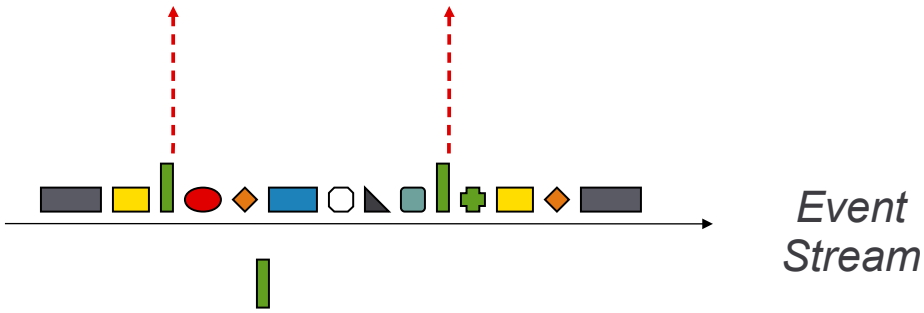


A union B



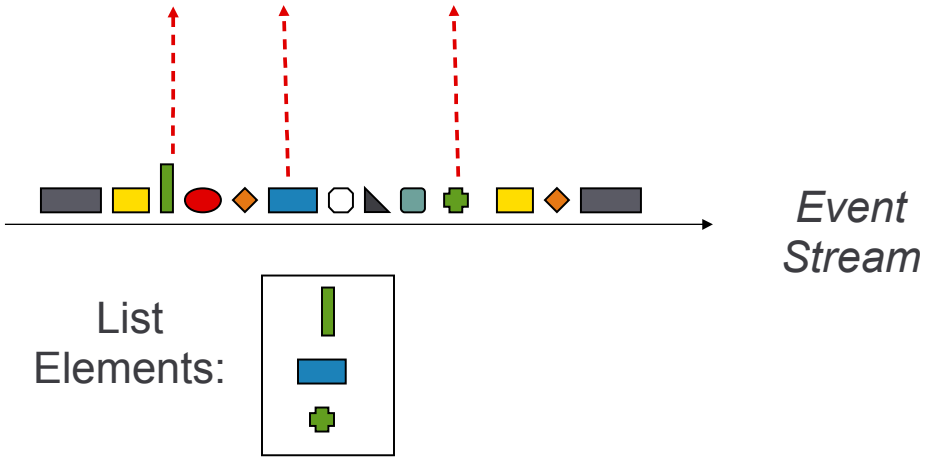
Filter Static Lookup

- Find an event with specific attributes
  - Severity 5 event from the Accounts server
    - filter(e.sev=5 and e.shn = “Accounts Server”)



Filter inList

- Severity 5 event from any critical server
  - filter(e.sev=5 and e.shn inlist CriticalServerList)



Filter inList

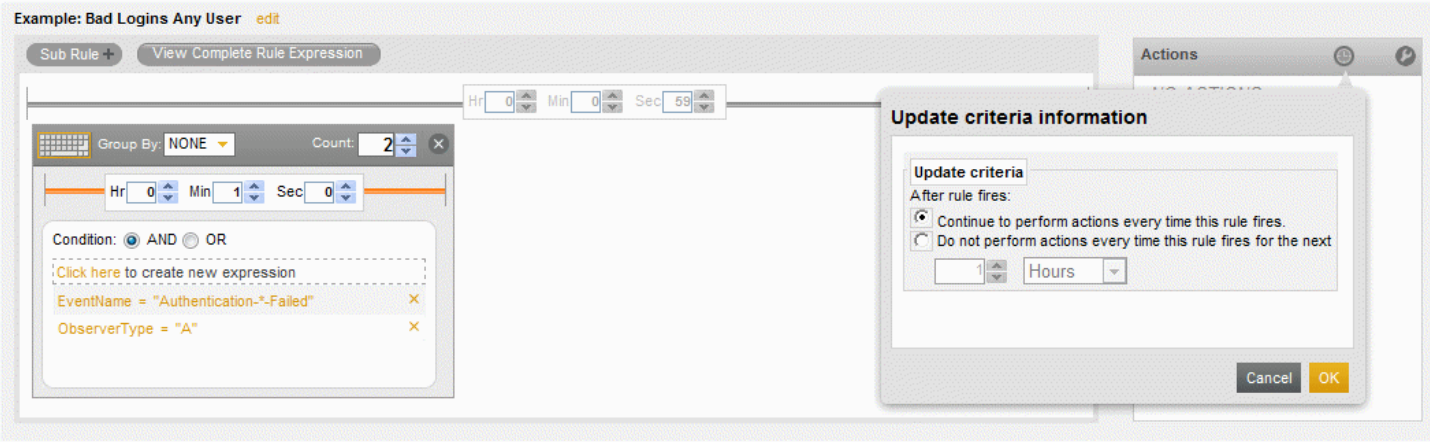
- Unauthorized SMTP
  - filter(e.st = “F” and not (e.sip inlist AuthorizedMailServerList) and e.dp=25)
- “inlist” operator used to check data within existing dynamic lists

Filter isNull

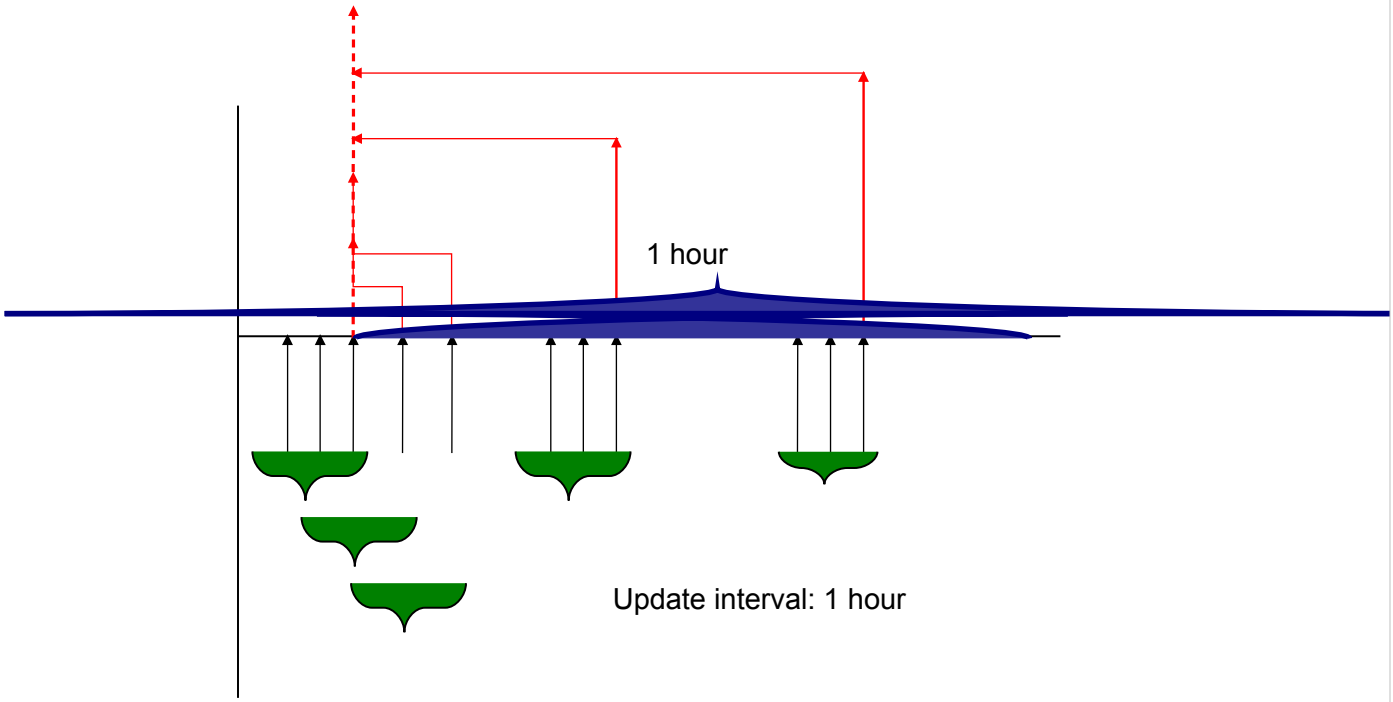
- Identify events with null attributes
- Always used within a filter
- Syntax
  - Filter(isNull(e.cv52) and e.evt="file access")
  - Filter(e.evt="file access" and not(isNull(e.cv52)))

Correlation Updates Interval

- Allows us to treat recurring patterns as a continuation of the same attack.
- Time Based
- User Defines Update interval



Time interval



filter(e.evt = “failed login”) flow trigger(3, 30)

Basic Correlation

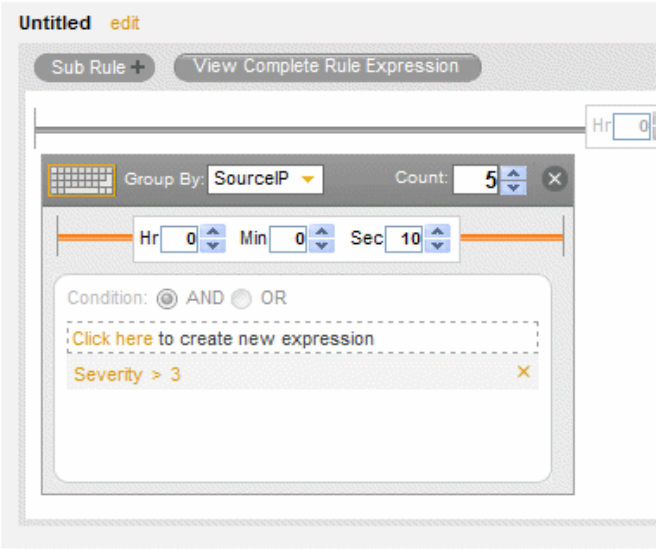
- We will start with basic functions and concepts (not meant to be used independently in a correlation)
- Next we will combine functions to make reasonable correlations
- Finally we will expand out Correlation vocabulary

RuleLG – Trigger Details

- **trigger(n,t)**
  - e.g. trigger(3, 90)
  - This operator will trigger when *n* meets the threshold of 3 keeping only *n-1=2* events in storage before triggering.
  - If *n=1*, then the operator will trigger immediately.
  - If *t=0*, the trigger is instantaneous; a special case used only for handling window output (see below)
- **trigger(n,t,discriminator(e))**
  - E.g. trigger(3,90,discriminator(e.sip))
  - This operator will trigger when *n* meets the threshold of 3 keeping *n-1=2* events in storage for each distinct Source IP

RuleLG – Trigger Example

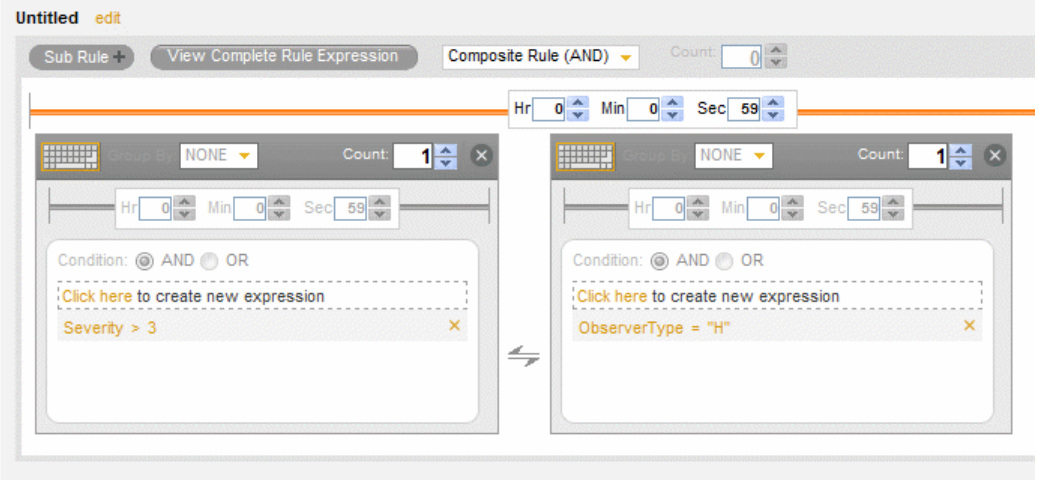
trigger(5,10, discriminator(e.SourceIP))
In this example, the trigger operation keeps distinct buckets for each unique Source IP; for each bucket, there is a count of 5 in 10 seconds.
If the threshold is met within the time frame, the trigger will generate a Correlated Event as an output set.



“Real-Life” Trigger RuleLG Example

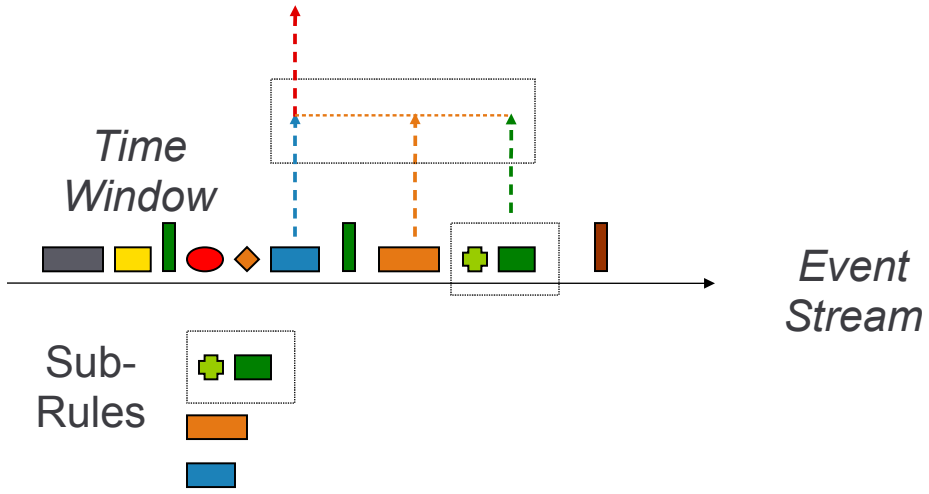
```
filter(e.DeviceCategory = "FW"
  and e.Severity >= 4) flow
trigger(5,600,discriminator(e.Des
      tinationHostName))
```

Composite Rule



Gate

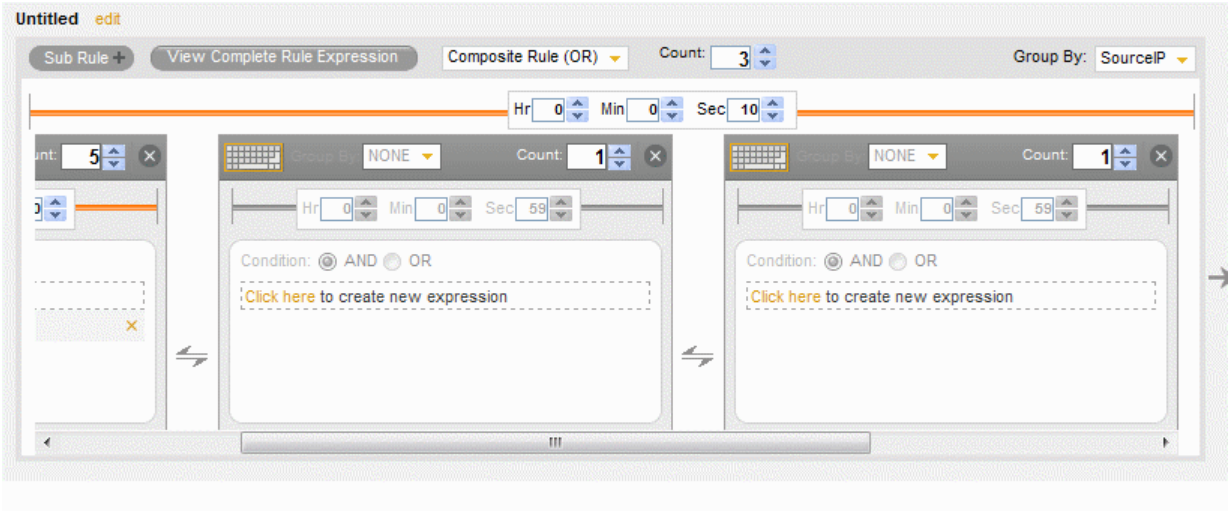
- Gate is composed of other rules
  - Sub-rule can be any rule, including another composite rule.
  - Rule fires if sub-rules fire within specified time period.
    - Configurable Mode: All, Any, Some (3 of 5)



Gate

- Syntax
  - gate(rule1, rule2, rule3...., <mode>, <time>, <discriminator>)
- Example
  - User logs in, changes password and withdraws more than 10,000 within a specified time frame.
    - gate(filter(e.evt="login"), filter(e.evt="passwordchange"), filter(e.evt="withdraw" and e.amt > 10000), all, 10m, discriminator(e.useraccount))

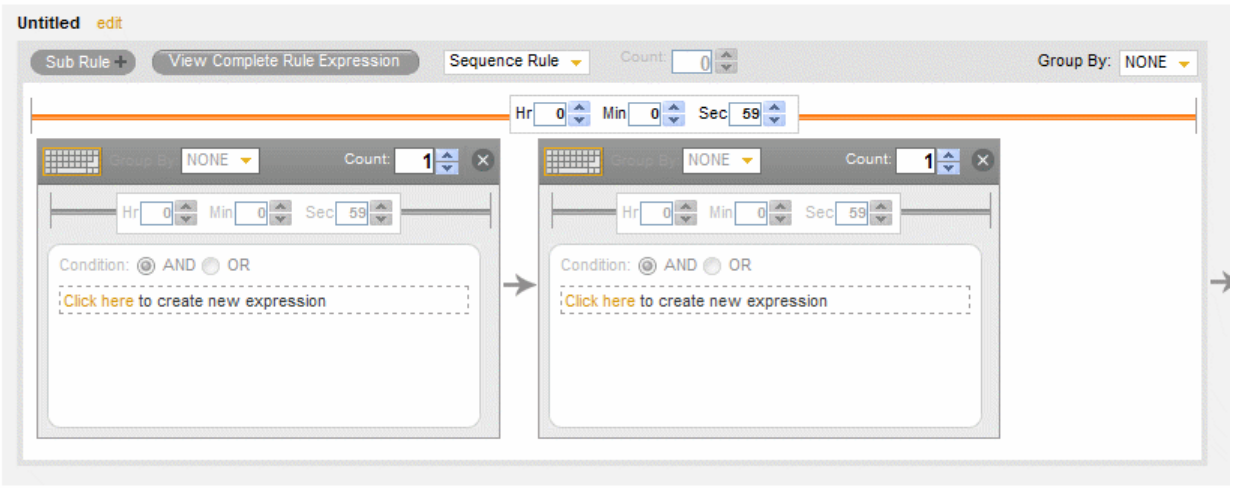
Composite Rule



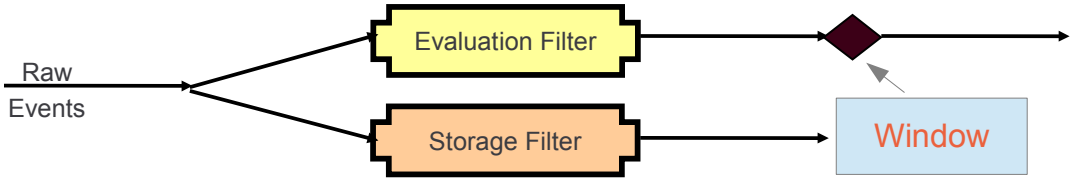
Sequence

- Complex pattern based on time-ordered sub-patterns.
- Rule fires only if all sub-rules fire in time order
- Syntax
  - sequence(rule1, rule2, rule3..., <time>, <discriminator>)
- Example
  - 3 failed logins followed by 1 successful login by the same user within 3 minutes.
    - sequence(filter(e.evt="login failed") flow trigger (3,180), filter(e.evt = "login"), 180, discriminator(e.sun))

Sequence



RuleLG – Window Filters



The window operator has two paths:

Evaluation path -

Input into the **window()** function, by default all events but can be pre-filtered using a standard **filter()** function.

Each incoming event is compared to events stored in the window by using the **Evaluation expression** (e.g.: w.sip = e.sip).

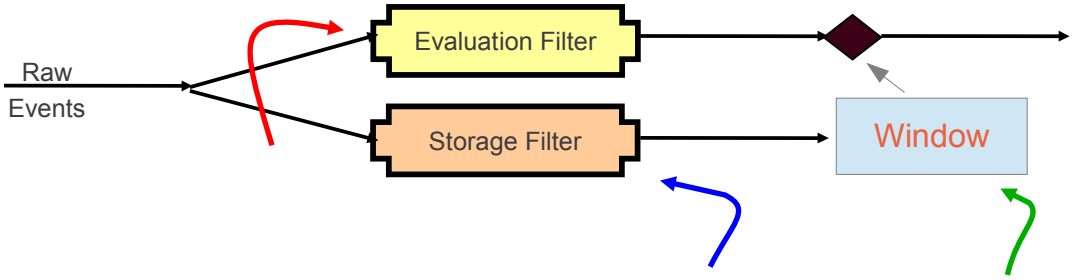
Storage path –

Determined by the **storage filter.**

A parallel stream of all raw events (but delayed by one event!) goes through the Storage Filter;

events that pass through the filter are added to the window.

RuleLG – Window Filters



The window operator has two paths:

Evaluation path `e.eventtag`

Stored bucket `w.eventtag`

Note that these two filters do **not** need to be the same. For example:

```
filter(e.rv32="IDS") flow
window (w.dip = e.dip, filter(e.rv32="FW"),
                500) flow
trigger(5,3600)
```

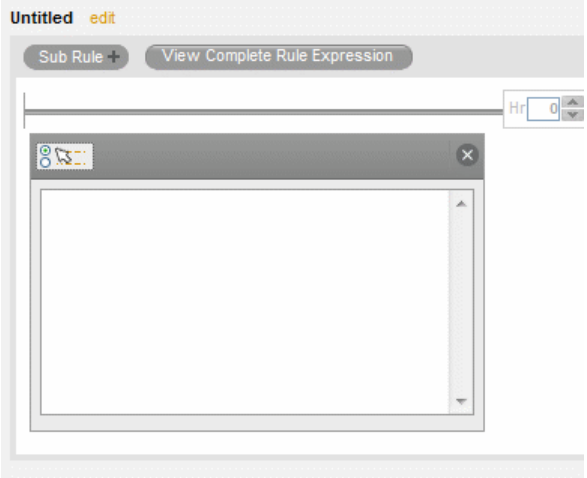
Major event from Firewall to the same Destination Host

```
filter(e.DeviceCategory = "FW"
    and e.Severity >= 4)
```

flow

```
trigger(5,600,discriminator
(e.DestinationHostName))
```

Free Form RuleLG Features



RuleLG – Window

```
window(w.sip=e.sip, filter(e.sip
                match
subnet(<192.168.12.0/24>)) ,60)
```

If the current event has a Source IP address that matches a past event's Source IP address that happened within the past 60 seconds, with the past events limited to those whose source IP is within the specified subnet, a correlated event is sent with the current event.

Window Example

Example:

```
filter(e.sev=2) flow window
(w.sip = e.sip, filter(e.sev=2),
500) flow trigger(5,3600)
```

“Real-Life” RuleLG Example

```
   filter(e.DeviceCategory = "FW" and
e.Severity >= 4)  flow window(w.SourceIP
 = e.SourceIP,filter(e.DeviceCategory =
  "FW" and e.Severity >= 4),120) flow
trigger(1,120,discriminator(e.SourceIP))
```

Major event from Firewall tracking SourceIP

filter(e.DeviceCategory = "FW" and e.Severity >= 4)

flow

window(w.SourceIP = e.SourceIP,filter(e.DeviceCategory = "FW" and e.Severity >= 4),120)

flow

trigger(1,120,discriminator(e.SourceIP))

Case Study

Case Study

This section presents several real world case studies.

They will involve:

A story describing what is happening

A control describing what the the company whats from Sentinel 7

An explanation about the security policy

The rule

And an observation about the rule.

Case Study 1 The Story

The Story:

In order to comply with regulations, the organization has to secure all the workstations to avoid access to classified information. is section presents several real world case studies.

One of the controls is to configure all terminals and sessions to block further authentication attempts, for a period of no less than 10 minutes, after 3 consecutive failed attempts from a user to login to his/her terminal.

Case Study 1 The Control

The Control:

The organization decides to use Sentinel 7 to monitor non compliant behavior through the use of correlations.

Specific to SSH in SUSEis section presents several real world case studies.

Case Study 1 The Explanation

The Explanation:

According to the organizations security policy, if all terminals are configured to block further authentication attempts after 3 failed attempts, is a fourth attempt is seen within 10 minutes if the first one, the workstation or terminal has not been configured correctly and is therefore non compliant.

Case Study 1 The Rule

The Rule:

A correlation rule that could be used to make sure that workstations and terminals are compliant might look like this.

filter(((e.EventName = “sshd: Authentication denied”)))flow trigger (4,600,discr4iminator(e,InitiatorUserName)

Case Study 1 An Observation

An Observation:

The previous rule applies specifically to SSH.

It would be more flexible to use taxonomy is possible.

Like the following example:

filter(e.XDASTaxonomyName=XDAS_AE_AUTHENTICATE_ACCOUNT AND (XDASOutcomeNAME=”XDAS_OUT_DENIAL” OR XDASOutcomeName=”XDAS_OUT_FAILURE”))flow trigger(4,600,descriminator(e.InitiatorUserName))

Case Study 2 The Story

The Story:

A system administrator at the bank is worried that external administrators from his service provider are abusing their privileged rights.

Upon investigation, he has found out that in order to hide their tracks, the external administrators would create a privileged user account with fake user details to avert suspicion.

Since all privileged user monitoring depends on filtering the KNOWN administrator accounts, the new account created would circumvent the controls because the username was not included in any of the monitoring lists.

Case Study 2 The Control

The Control:

To find out who else is creating fake privileged users and in order to monitor this illicit behavior, the security administrator uses Sentinel 7 to correlate the deviant behavior.

His investigation saw that the administrator would create the user and then use another account to delete it.

Case Study 2 The Rule

The Rule:

Based on the control, the security administrator create the following rule.

sequence(filter(((e.XDASTaxonomyName = “XDAS_AE_CREATE_ACCOUNT”))),filter(((e.XDASTaxonomyName = “XDAS_AE_DELETE_ACCOUNT”))) , 3600,discriminator(e.TargetUserName))

Case Study 2 An Observation

An Observation:

There may be a reason to increase the evaluation time period to more than just one hour.

If that were the case, the correlation rule rule should be run on a separate correlation server.

Case Study 3 The Story

The Story:

The organization has a development environment where application are constantly being tested.

This environment has been segregated from the rest of the production network and is treated as an UNTRUSTED segment of the network.

The auditor has asked to monitor the communications between this network and the production subnet.

Case Study 3 The Control

The Control:

The administrator is using Sentinel 7 correlations to monitor non compliant behavior from the development network.

The development network is 172.16.0.0/24

Case Study 3 The Rule A

The Rule:

Repetitive illegal communication detected by the firewall, usually indicates a worm wanting to break out.

This is more of an early threat detector.

filter(e.DeviceCategory = “FW” and e.XDASTaxonomyName = “XDAS_AE_TERMINATE_PEER_ASSOC” and e.XDASOutcomeName = “XDAS_OUT_DENIAL” and ((e.sip match subent(172.16.0.0/24))))flow trigger(1,60,descriminator(e.sip,e.dip))

Case Study 3 The Rule B

The Rule:

This rule is configured to do no action for 3 hours.

Every correlation event will contain non compliant communication between IP pairs.

filter(e.DeviceCategory = “FW” and ((e.sip match subent(172.16.0.0/24))) and ((e.dip match subnet(192.168.0.0/16))))flow trigger(1,60,descriminator(e.sip,e.dip))

Case Study 3 The Rule C

The Rule:

This rule is meant to alert on the non compliant behavior of using production logs to test systems by filtering known ports belonging to auditing and logging applications and DB ports.

Also configured to not do anything for X amount of hours.

filter(e.DeviceCategory = “FW” and (e.dp=512 or e.dp=1512 or e.dp=289 or e.dp=1289 or e.sp=1524 or sp=e.3306) and ((e.sip match subent(192.168.0.0/16))) and ((e.dip match subnet(172.16.0.0/24))))flow trigger(1,60,descriminator(e.sip,e.dip))

Lab exercise 1, 2, and 3

Lab exercise 1 will have you use a built in correlation rule.

Lab exercise 2 will have you create a simple correlation rule.

Lab exercise 3 will have you build the correlation rules associated with these case studies.