



# Amazon Fraud Detector

Amazon Fraud Detector is a fully managed machine learning service by AWS that helps businesses identify and prevent online fraud in real-time. It simplifies the process of building and deploying machine learning models to detect suspicious activities like online payment fraud, fake account creation, and identity theft without requiring machine learning expertise.

## Key Features:

### 1. Fraud Detection Models:

- Amazon Fraud Detector provides pre-built fraud detection models trained on Amazon's own data from years of combating fraud on its e-commerce platform. These models are highly accurate and can detect fraudulent activities with minimal configuration.

### 2. Custom Model Training:

- Users can upload their own historical data to train custom fraud detection models. The service will automatically handle the complexities of model development, including feature engineering, model selection, and tuning.

### 3. Real-Time Predictions:

- Once deployed, Amazon Fraud Detector evaluates events like account creation or transaction attempts and returns fraud risk predictions in real time, allowing businesses to take immediate actions (e.g., flagging suspicious accounts or blocking transactions).

### 4. Fraud Risk Scores and Outcomes:

- The service provides a fraud risk score for each event and allows users to define outcomes based on these scores. For instance, if the score is above a certain threshold, the system might trigger a manual review, send an alert, or block the transaction.

### 5. Customizable Rules Engine:

- Users can combine machine learning-based fraud predictions with custom business rules to better fine-tune fraud detection according to their specific use cases. For example, they might flag all transactions from certain locations as higher risk.

### 6. Integration with AWS Services:

- Amazon Fraud Detector integrates seamlessly with other AWS services like Amazon S3 for data storage, AWS Lambda for triggering actions based on predictions, and Amazon CloudWatch for monitoring and logging.

### 7. Fraud Detection Use Cases:

- Common fraud detection scenarios include:

- **Online payment fraud:** Detect suspicious transactions in e-commerce or financial services.
- **Account takeovers:** Identify unusual login behavior or attempts to access accounts using stolen credentials.
- **Fake account creation:** Prevent the creation of fake or fraudulent accounts used for spamming or other malicious activities.
- **Loyalty fraud:** Detect suspicious behavior in loyalty or rewards programs.

#### **Benefits:**

- **No Machine Learning Expertise Required:** Businesses can easily implement fraud detection solutions without having to build complex machine learning models from scratch.
- **Pre-Trained Models:** Amazon's models are built from years of experience and data, so businesses benefit from a sophisticated fraud detection system right out of the box.
- **Customizable to Your Business:** You can customize the system to suit specific business needs by uploading your own data or defining business-specific fraud detection rules.
- **Scalable:** Fraud Detector scales automatically to handle large volumes of transactions and events.
- **Real-Time Detection:** Allows for instant decision-making on potentially fraudulent activities.

#### **How it Works:**

1. **Data Collection:** The business provides historical event data (such as previous transactions, account registrations, or fraud outcomes) to Amazon Fraud Detector.
2. **Model Training:** The service either uses pre-built models or trains custom models on the provided data to learn patterns indicative of fraud.
3. **Deploy and Monitor:** Once deployed, the model can evaluate real-time events and produce fraud risk scores. These scores are used to trigger actions, such as blocking transactions or flagging them for review.

#### **Integration:**

- **Amazon S3:** Store and retrieve your event data for model training.
- **Amazon SNS:** Notify teams or systems when potential fraud is detected.
- **AWS Lambda:** Automate actions based on fraud detection, such as disabling user accounts or flagging suspicious transactions.
- **Amazon CloudWatch:** Monitor and log the performance of fraud detection models.

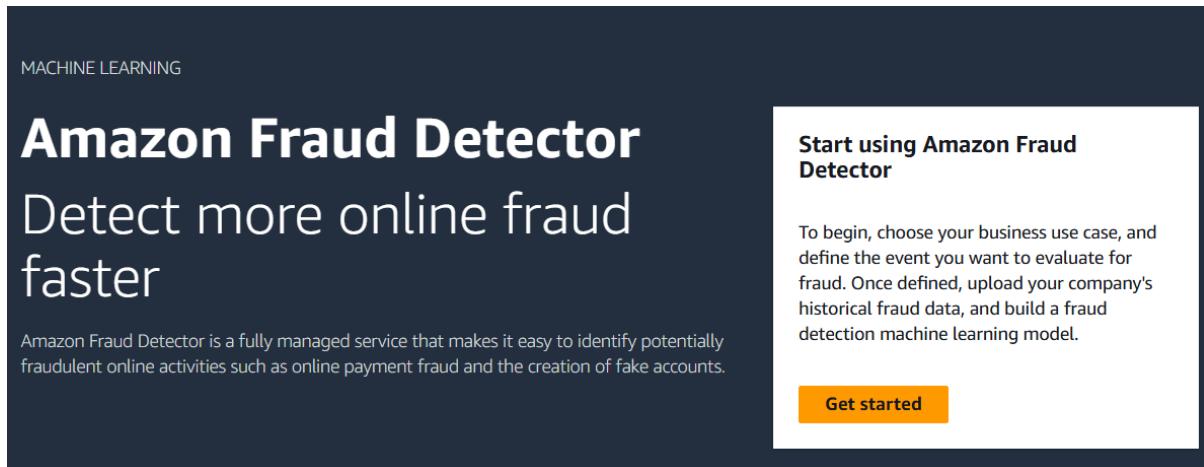
#### **Common Use Cases:**

- **Financial Services:** Detect fraudulent credit card transactions or suspicious account activity.
- **E-commerce:** Prevent payment fraud and protect customer accounts from hacking attempts.
- **Insurance:** Identify fraudulent claims or attempts to exploit insurance systems.
- **Gaming:** Detect fraudulent behavior in gaming platforms such as fake accounts or cheating.

In summary, Amazon Fraud Detector provides a highly effective, scalable, and easy-to-use solution for detecting and mitigating fraud using machine learning. It helps businesses protect their operations and customers by reducing the likelihood of fraud losses.

## To begin with the Lab:

1. In your AWS Console, search for Amazon Fraud Detector and navigate to it. Click on Get started. Just remember that the region used in this service is Singapore.



2. So, to begin with the fraud detector from the left pane choose events and click on create event type.

Name	Description	Entity types	Date created
No results			
No results match your current search criteria.			
<a href="#">Create event type</a>			

3. Here you need to give it a name and description then choose to create a new entity.

## Event type details

With Amazon Fraud Detector, you generate fraud predictions for events. An event type defines the structure for an event sent to Amazon Fraud Detector. This includes the variables sent as part of the event, the entity performing the event (such as a customer), and the labels that classify the event. Example event types include online payment transactions, account registrations, and authentications. Once defined, you can use models and detectors to evaluate the fraud risk for an event.

Name

Event type name must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description

Entity

Select the entity for this event. An entity represents who is performing the event. Example entities include customer, merchant, or account.

4. Below you can see that our entity has been created.

## Create event type

### Event type details

With Amazon Fraud Detector, you generate fraud predictions for events. An event type defines the structure for an event sent to Amazon Fraud Detector. This includes the variables sent as part of the event, the entity performing the event (such as a customer), and the labels that classify the event. Example event types include online payment transactions, account registrations, and authentications. Once defined, you can use models and detectors to evaluate the fraud risk for an event.

Name

Event type name must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description

Entity

Select the entity for this event. An entity represents who is performing the event. Example entities include customer, merchant, or account.

5. Then in the event variables section you need to choose the event highlighted below in the snapshot and then choose to create a new IAM role.

## Event variables

Each event type is represented by a collection of related variables.

Choose how to define this event's variables

Select variables from a training dataset

### IAM role

Amazon Fraud Detector requires permission to access datasets contained within S3 buckets. Choose a role or let us create a role with the AmazonFraudDetector-DataAccessPolicy IAM policy attached. If you created a new role to access this data, please wait for 30 seconds after role creation before proceeding.

Select an IAM role

Create IAM role

Enter a custom role A

S3 location must be located in your current Fraud Detector region. Your file must be in CSV format. Example: s3://bucket/my-training-dataset.csv

6. Here you can see that it is saying that you need to provide the name of your bucket.
7. Here you need to create an S3 bucket with a unique name and then upload the training data which you can get from the AWS GitHub repository using the link given down below.

[https://github.com/aws-samples/aws-fraud-detector-samples/blob/master/data/registration\\_data\\_20K\\_full.csv](https://github.com/aws-samples/aws-fraud-detector-samples/blob/master/data/registration_data_20K_full.csv)

Amazon S3 > Buckets > demo-fd-bucket-2

demo-fd-bucket-2 [Info](#)

Objects Properties Permissions Metrics Management Access Points

Objects (1) [Info](#)

Actions ▾ Create folder Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

Name	Type	Last modified	Size	Storage class
<a href="#">registration_data_20K_full.csv</a>	csv	October 21, 2024, 14:52:28 (UTC+05:30)	4.2 MB	Standard

## Create IAM role

X

Creating an IAM role gives Amazon Fraud Detector permission to read files in your specified S3 buckets so that it can generate predictions using your datasets.

The IAM role you create will grant Amazon Fraud Detector access to the following buckets to read input files and store output files. If you do not plan to store output files in a separate bucket, enter the same bucket name for both.

demo-fd-bucket-2

Use bucket names only; do not include s3://. Separate multiple bucket names using commas. ARNs, "\*", and "/" are not supported.

Cancel

Create role

- Now here you need to provide the Data location of your object so from your S3 bucket you need to copy the S3 object URI and paste it here then **click on Upload**.

### Event variables

Each event type is represented by a collection of related variables.

Choose how to define this event's variables

Select variables from a training dataset

#### IAM role

Amazon Fraud Detector requires permission to access datasets contained within S3 buckets. Choose a role or let us create a role with the AmazonFraudDetector-DataAccessPolicy IAM policy attached. If you created a new role to access this data, please wait for 30 seconds after role creation before proceeding.

AmazonFraudDetector-DataAccessRole-1729502655813



Success! You created an IAM role.

AmazonFraudDetector-DataAccessRole-1729502655813

#### Data location

Provide the S3 location of your data. [Go to S3 to copy the path to your dataset location](#)

s3://demo-fd-bucket-2/registration\_data\_20K\_full.csv

Upload

S3 location must be located in your current Fraud Detector region. Your file must be in CSV format. Example: s3://bucket/my-training-dataset.csv

- After click on Upload, you will see that you have a new section for Variable and variable types. So, fill the same variable types as you can see below.

Variable (7)	Variable type		
ip_address	IP Address	▼	Remove
email_address	Email Address	▼	Remove
billing_state	Billing Address: State or Province	▼	Remove
user_agent	User Agent	▼	Remove
billing_postal	Billing Address: Zip Code	▼	Remove
phone_number	Phone Number	▼	Remove
billing_address	Billing Address: Address Line 1	▼	Remove

10. Then in the labels you need to create two labels for fraud and legit as you can see below.
11. After that move forward and click on create event.

**Labels - optional**

To train an ML model using this Event, you must define at least two labels. Labels are used to categorize individual events as either fraud or legitimate using any labels you define.

Labels

Choose labels ▾

fraud X   legit X

12. So, once your event has been created now you need to create a model for it.
13. From the left pane choose models and click on add model.

Fraud Detector > Models

## Models

Add model ▾

Models (0)	Fraud Detector models	SageMaker models
------------	-----------------------	------------------

Welcome to the Amazon Fraud Detector model library

Machine learning models can help increase the efficiency and accuracy of risk classification within your organization. Start building your model library by configuring your first model, all without writing a single line of code.

Add your first model

14. Here you need to give it a name and scroll down to event type and choose your recently created event.

# Define model details

## Model details

Model name

Model names must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description - *optional*

Model type



Event type



or [create a new event type](#).

15. In the historical event data, you need to choose event data stored in S3. For the IAM role you need to create a new role.

## Historical event data

### Event data source

Select the source for events data that will be used to train this model.

- Event data stored in S3
- Event data stored in Amazon Fraud Detector

To train the model, upload a CSV of historical data representing events of the selected type. The dataset must include the reserved headers EVENT\_TIMESTAMP and EVENT\_LABEL and at least two of the variables defined in the event type. For details on how to build a training dataset, refer to the documentation [\[2\]](#).

### IAM role

Amazon Fraud Detector requires permission to access datasets contained within S3 buckets. Choose a role or let us create a role with the AmazonFraudDetector-DataAccessPolicy IAM policy attached. If you created a new role to access this data, please wait for 30 seconds after role creation before proceeding.

Select an IAM role

### Training data location

Provide the S3 location of your data. [Go to S3 to copy the path to your dataset location \[2\]](#)

### Data Location

s3://bucket/prefix/object

[View](#)

[Browse S3](#)

S3 location must be located in your current Fraud Detector region. Your file must be in CSV format. Example: s3://bucket/my-training-dataset.csv

16. You need to create this role in the similar manner as you did before, give it your S3 bucket name and click on create role.

## Create IAM role



Creating an IAM role gives Amazon Fraud Detector permission to read files in your specified S3 buckets so that it can generate predictions using your datasets.

The IAM role you create will grant Amazon Fraud Detector access to the following buckets to read input files and store output files. If you do not plan to store output files in a separate bucket, enter the same bucket name for both.

demo-fd-bucket-2

Use bucket names only; do not include s3://. Separate multiple bucket names using commas. ARNs, "\*", and "/" are not supported.

[Cancel](#)

[Create role](#)

17. In the data location you need to give the S3 object URI then click on Next.

#### IAM role

Amazon Fraud Detector requires permission to access datasets contained within S3 buckets. Choose a role or let us create a role with the AmazonFraudDetector-DataAccessPolicy IAM policy attached. If you created a new role to access this data, please wait for 30 seconds after role creation before proceeding.

AmazonFraudDetector-DataAccessRole-1729503548565



Success! You created an IAM role.

AmazonFraudDetector-DataAccessRole-1729503548565

#### Training data location

Provide the S3 location of your data. Go to S3 to copy the path to your dataset location [↗](#)

#### Data Location

t-2/registration\_data\_20K\_full.csv [X](#)

[View ↗](#)

[Browse S3](#)

S3 location must be located in your current Fraud Detector region. Your file must be in CSV format. Example: s3://bucket/my-training-dataset.csv

18. Here you can see that you the model inputs.

## Configure training

### Model inputs

By default, Amazon Fraud Detector will use all variables from your historical dataset as model inputs. If you do not want to include certain variables, deselect them from the list below.

<input checked="" type="checkbox"/>	Variable	Variable type
<input checked="" type="checkbox"/>	billing_state	Billing Address: State or Province
<input checked="" type="checkbox"/>	billing_postal	Billing Address: Zip Code
<input checked="" type="checkbox"/>	email_address	Email Address
<input checked="" type="checkbox"/>	ip_address	IP Address
<input checked="" type="checkbox"/>	billing_address	Billing Address: Address Line 1
<input checked="" type="checkbox"/>	user_agent	User Agent
<input checked="" type="checkbox"/>	phone_number	Phone Number

19. Then in the label classification choose your labels and move to review page. Click on Create and train model.

## Label classification

Labels are used to categorize individual events as either fraud or legitimate using any labels you define.

### Fraud labels

Select one or more labels from the event type to categorize fraudulent events.

*Choose at least one label*

fraud 

### Legitimate labels

Select one or more labels from the event type to categorize legitimate events.

*Choose at least one label*

legit 

20. Below you can see that our model has been created and currently it is being trained and it will take a lot of time so you need to wait for it.

fd\_model

Actions ▾

#### Model details

Edit

Model name	Model type	Date created
fd_model	Online Fraud Insights	Now
Description	Event type	ARN
-	demo-fd	 arn:aws:frauddetector:ap-southeast-1:533267094905:model/ONLINE_FRAUD_INSIGHTS/fd_model

#### Model versions (1)

Version	▲	Performance (AUC)	▼	Date created	▼	Last updated	▼	Status	▼
1.0	-			Now		Now		 Training...	

21. Below you can see that our model has been trained and it is ready to deploy. If you click on version to open it.

Fraud Detector > Models > fd\_model

## fd\_model

**Model details**

Model name fd_model	Model type Online Fraud Insights	Date created 44 minutes ago
Description -	Event type demo-fd	ARN <a href="#">arn:aws:frauddetector:ap-southeast-1:533267094905:model/ONLINE_FRAUD_INSIGHTS/fd_model</a>

**Model versions (1)**

Version	Performance (AUC)	Date created	Last updated	Status
1.0	0.95	44 minutes ago	10 minutes ago	Ready to deploy

22. In this model version you can see the model performance.

### Model performance

Model type: Online Fraud Insights  
**AUC: 0.95 (Uncertainty range: 0.94–0.96)**

**Score distribution**

By writing a rule using a model score threshold of **500**, you will succeed in catching **93.2%** of all fraudulent events (TPR) while accepting a risk that **13.7%** of legitimate events are incorrectly labeled as fraud (FPR).

Click anywhere on the chart above to select a model threshold score and determine the TPR and FPR.

**Confusion matrix**

For the selected model score threshold, the confusion matrix represents the expected outcome given 100,000 sample events (95,189 legitimate, 4,811 fraud).

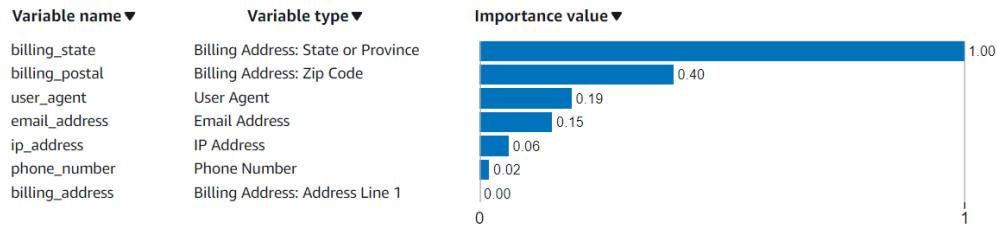
		Predicted		
		Fraud	Legitimate	
Actual	Fraud	<b>True positive 4454</b>	False negative 324	TPR 93.2%
	Legitimate	False positive 13072	<b>True negative 82150</b>	FPR 13.7%

Numbers are based on a sample of 100,000 events.

23. The model variable importance and other things.

## Model variable importance

Variable importance gives you an understanding of how different variables are contributing to your model's performance. The chart below lists variables in the order of its importance to the model, indicated by the number. Variables of AGGREGATE variable type are a combination of multiple variables that are enriched and have an aggregated importance value. A variable (raw or aggregate) with a much higher number relative to the rest could indicate that the model might be overfitting on it, while variables with relatively lowest numbers could just be noise. For more information, see documentation [see documentation](#)



24. Now scroll up and click on Actions and choose to deploy model version.

Fraud Detector > Models > fd\_model > Version 1.0

### fd\_model (Version 1.0)

[Overview](#) | [Configuration](#)

**Actions**

- Deploy model version
- Delete
- Cancel training

**Version details**

Status	Event type	Output variable	Date created
⌚ Ready to deploy	demo-fd	fd_model_insightscore	49 minutes ago
ARN	arn:aws:frauddetector:ap-southeast-1:533267094905:model-version/ONLINE_FRAUD_INSIGHTS/fd_model/1.0		

25. Click on deploy version and it will take at least 10 minutes to deploy this model.

## Deploy model version



Deploying this model will make it available to add to detectors for real-time fraud predictions. You will be charged by the hour for on-demand hosting of your deployed model.

[Cancel](#)

[Deploy version](#)

26. Below you can see our model is active now.

Fraud Detector > Models > fd\_model > Version 1.0

## fd\_model (Version 1.0)

Actions ▾

Overview Configuration

### Version details

Status	Event type	Output variable	Date created
Active	demo-fd	fd_model_insightscore	1 hour ago
ARN	arn:aws:frauddetector:ap-southeast-1:533267094905:model-version/ONLINE_FRAUD_INSIGHTS/fd_model/1.0		

27. Now we need to create a detector. Click on Create detector.

### Detectors (0)

Detectors are comprised of models and rules that evaluate events for fraud.

Find detectors < 1 > ⚙

Detector name	Description	Event type	Date created
No results			
No results match your current search criteria.			
<a href="#">Create detector</a>			

28. First you need to give it a name, then choose your event type and click on next.

## Define detector details

### Detector details

Detector name

Detector names must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description - *optional*

Event type

Select the type of event to be evaluated for fraud



or [create a new event type](#).

### ► Detector tags - *optional*

Tags are key-value pairs that you can add to AWS resources to help identify, organize, and search for resources.

Cancel

Next

29. On the next page you need to click on add model and choose your model and version number. Then click on next.

## Add model - *optional*

Add one or more models to this detector version to help detect fraudulent events. To add a model, it must be configured to evaluate the selected event type: demo-fd.

If you haven't made a model yet, [go back to the model library](#), configure a new model, and train it. Once it's been trained, you can assess its performance and use it with different detectors.

Fraud Detector models (0)

SageMaker endpoints (0)

Remove model

Add model

Model name



Version



Model output variable



Status



No custom models selected

Cancel

Previous

Next

## Add model



Fraud Detector models

SageMaker models

### Model

Select an Amazon Fraud Detector model and version. The model version must finish deploying and be active to be selected.

fd\_model

Online Fraud Insights ▾

1.0 ▾

Cancel

Add model

30. In the add rules section you need to write a name for your rule and in the expression, you need to give the same expression as you can see below. You just need to write \$ sign and you will have the expression.

## Add rules

### Define a rule

Rules are made of conditions and actions. If the condition is detected on an incoming event, the action(s) will trigger.

Name

fraud\_rule

Version

1

Rule names must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description - optional

Describe what this rule monitors

### Expression

Using Amazon Fraud Detector's simplified expression language, you can write rules to evaluate event variables or model output scores. To reference these variables, type "\$" to start searching the variables library. Use the expression quick reference guide below for help.

1    \$fd\_model\_insightscore>900

31. Then you need to create a new outcome. For the outcome name write risk\_high and click on save outcome. In the end click on add rule.

## Outcomes

What should happen when the condition above is met?

Choose one or more outcomes...

Create a new outcome

► **Tags - optional**

Tags are key-value pairs that you can add to AWS resources to help identify, organize, and search for resources.

Add rule

## Create a new outcome

X

Outcome name

risk\_high

Outcome names must be a-z, all lowercase characters, no spaces (underscores are allowed).

Outcome description

Add a description

► **Tags - optional**

Tags are key-value pairs that you can add to AWS resources to help identify, organize, and search for resources.

Cancel

Save outcome

32. Then you need to click on add another rule.

## Add rules

Name  
fraud\_rule

Version  
1

Outcomes  
risk\_high

Edit

Remove

► View expression

Add another rule

Cancel

Previous

Next

33. Choose create rule and give your rule a name and in the expression give the same as shown below.

[Use existing rule](#)      **Create rule**

---

**Define a rule**

Rules are made of conditions and actions. If the condition is detected on an incoming event, the action(s) will trigger.

Name	Version
legit_rule	1

Rule names must be a-z, all lowercase characters, no spaces (underscores are allowed).

Description - *optional*

*Describe what this rule monitors*

**Expression**

Using Amazon Fraud Detector's simplified expression language, you can write rules to evaluate event variables or model output scores. To reference these variables, type "\$" to start searching the variables library. Use the expression quick reference guide below for help.

```
1 $fd_model_insightscore<700
```

34. In the outcome again you need to create a new outcome and this time give the outcome name as risk low. Then click on Add rule.

**Outcomes**

What should happen when the condition above is met?

*Choose one or more outcomes...*

risk\_low X

► **Rule tags - optional**

Tags are key-value pairs that you can add to AWS resources to help identify, organize, and search for resources.

[Cancel](#)

[Add rule](#)

35. So, we have created two rules one is for fraud and other one is for legit. Now we are going to create another rule for review. So, click on add another rule.
36. Choose to create a new rule give it a name and then give the same expression as shown below.

Use existing rule
Create rule

---

### Define a rule

Rules are made of conditions and actions. If the condition is detected on an incoming event, the action(s) will trigger.

Name	Version
review_rule	1

Rule names must be a-z, all lowercase characters, no spaces (underscores are allowed).

**Description - optional**

*Describe what this rule monitors*

**Expression**

Using Amazon Fraud Detector's simplified expression language, you can write rules to evaluate event variables or model output scores. To reference these variables, type "\$" to start searching the variables library. Use the expression quick reference guide below for help.

```
1 $fd_model_insightscore<900 and $fd_model_insightscore>700
```

37. Then in the outcomes you need to create a new outcome as shown below and click on add rule.

**Outcomes**

What should happen when the condition above is met?

*Choose one or more outcomes...*

risk\_medium
X

**Rule tags - optional**

Tags are key-value pairs that you can add to AWS resources to help identify, organize, and search for resources.

Cancel
Add rule

38. Below you can see that we have 3 rules added. Now click on next and move to review page and create your detector.

## Add rules

Name  
fraud\_rule

Version  
1

Outcomes  
risk\_high

Edit

Remove

► View expression

Name  
legit\_rule

Version  
1

Outcomes  
risk\_low

Edit

Remove

► View expression

Name  
review\_rule

Version  
1

Outcomes  
risk\_medium

Edit

Remove

► View expression

Add another rule

Cancel

Previous

Next

39. We need to give the values in the run test area so that we can test our dataset.

## Run test

To test the outcome of this version, provide values for each variable below derived from the version's ruleset. Once you have added all the relevant values run the test to see if the version results in the expected outcome. If there are default values for variables, they will be autopopulated below. The returned outcomes will be based on the detector rule version's rule execution type, either all matched rules' outcomes or the first matched rule's outcome(s).

### Event metadata

Timestamp

2024/10/21



00:00:00

EntityId

unknown

### Event variable

### Value

billing\_address

*Input value*

Default value

billing\_postal

*Input value*

Default value

billing\_state

*Input value*

Default value

email\_address

*Input value*

Default value

ip\_address

*Input value*

Default value

phone\_number

*Input value*

Default value

user\_agent

*Input value*

Default value

40. Here you can see that we have filled all the values. And you can get these values from the CSV file you download and uploaded to your S3 bucket at the start of this lab.  
41. Now click on run test.

Event variable	Value
billing_address	12351 Amanda Knolls Fake St.
billing_postal	34491
billing_state	NC
email_address	fake_cgonzales@example.net
ip_address	112.136.132.151
phone_number	(555)333 - 9246
user_agent	Mozilla/5.0 (iPad; CPU iPad OS 10_3_3 like Mac OS X) AppleWebKit/604.1.38 (KHTML, like Gecko) Version/10.3.3 Mobile/14E269 Safari/604.1

**Run test**

42. Below you can see that we get the outcome as risk low which means that it is legit event and its score is 153 which is less than 700.

 **Outcome: risk\_low**

**Model scores**

**fd\_model\_insightscore: 153**

**Run test**

43. Now we have taken the information for a fraud data and filled it here.

Event variable	Value
billing_address	65898 Amy Estate Fake St.
billing_postal	32931
billing_state	AZ
email_address	fake_timothysmith@example.com
ip_address	59.157.144.1
phone_number	(555)596 - 5579
user_agent	Mozilla/5.0 (iPod; U; CPU iPhone OS 3_0 like Mac OS X; sid-

44. And here you can see that we get the outcome as risk high which means that it is a fraud event whose event score is 997 which is more than 900 of value.

 **Outcome: risk\_high**

**Model scores**

fd\_model\_insightscore: 997

45. Once you are done with this lab just delete all the things.