

Cloud Security - Final Project (Write Up Form)

Requires an AWS account to perform the listed requirements:

- AWS Services:
 - Amazon Lex (including Analytics)
 - Lambda
 - Simple Notification Service (SNS)
 - CloudWatch

Amazon Lex:

This chatbot was created with the intention of having an interactive conversation through text.. I had to create a bot,add my intents, and include my custom slots with my intents to represent an action based on what the user wants to perform.

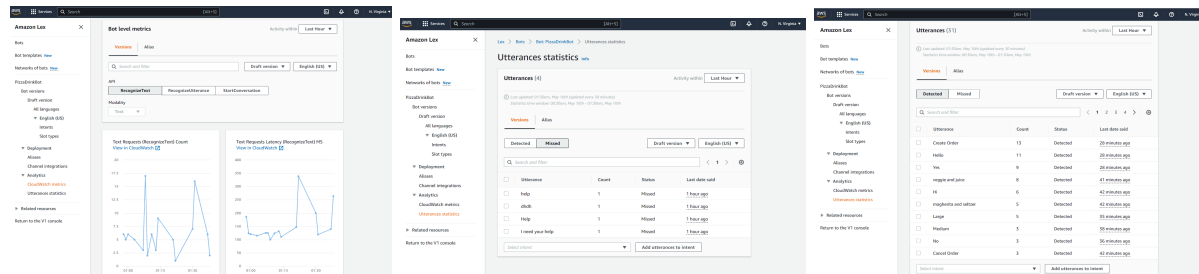
The screenshot shows the Amazon Lex console interface. On the left is a navigation sidebar with options like 'Bots', 'Bot templates', 'Networks of bots', and 'PizzaDrinkBot'. The main area displays the 'Intents' page for the 'PizzaDrinkBot' in the 'English (US)' language. It shows a list of four intents: 'CreateOrderIntent', 'WelcomeIntent', 'FallbackIntent', and 'CancelOrderIntent'. Each intent has a description and a 'Last edited' timestamp of '1 hour ago'. The 'FallbackIntent' description is 'Default intent when no other intent matches'. At the top of the main area, there are buttons for 'Draft version', 'English (US)', 'Successfully built', 'Build', and 'Test'.

Name	Description	Last edited
CreateOrderIntent	-	1 hour ago
WelcomeIntent	-	1 hour ago
FallbackIntent	Default intent when no other intent matches	1 hour ago
CancelOrderIntent	-	1 hour ago

Here is an example of the chatbot conversation for ordering a pizza with a drink:

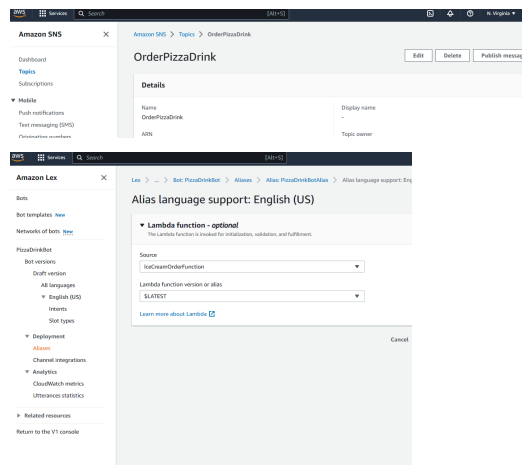
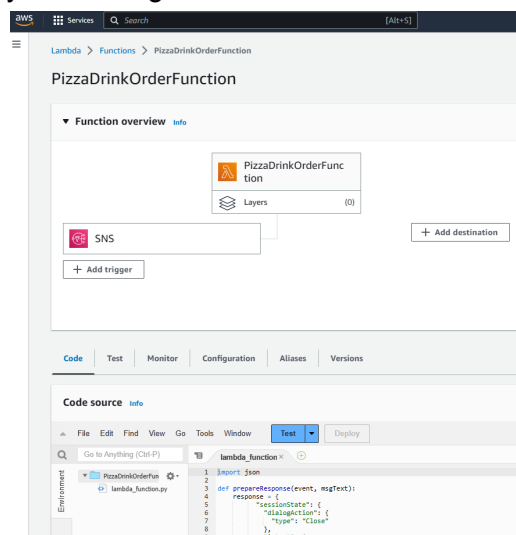
The four screenshots show a sequence of chatbot interactions. The first screenshot shows the initial 'Hello' message and a 'Create Order' button. The second screenshot shows the user's name 'Kelly' and the prompt 'What pizza and drink category will you choose?'. The third screenshot shows a list of options: 'Veggie and Juice', 'Margherita and Seltzer', 'Pepperoni and Soda', and 'Italian Sausage and Wine'. The user selects 'magheritaseltzer'. The fourth screenshot shows the prompt 'What size- Small/Medium/Large?' and the user selects 'Large'. The final message confirms the order: 'Your Order for, Large magheritaseltzer Pizza and Drink have been placed with Order#: 584892'.

You can also view the analytics of these conversations in CloudWatch metrics (i.e. graphs) and Utterances statistics (i.e. missed and detected) under Amazon Lex.



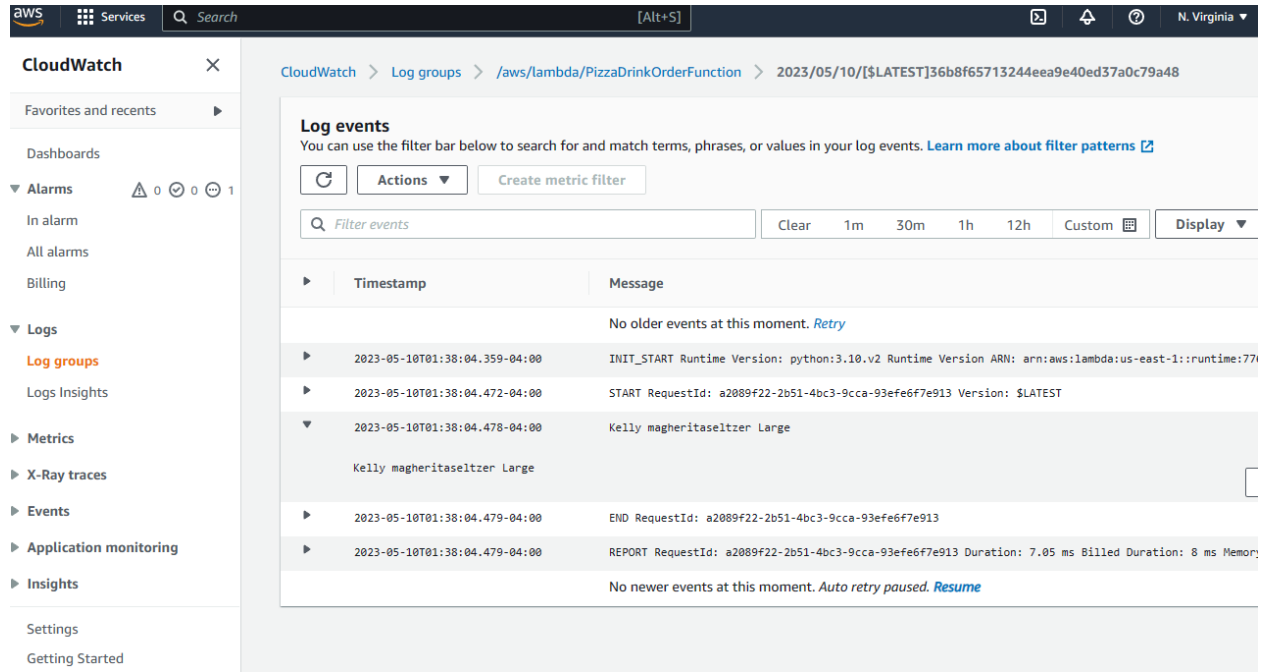
Lambda and SNS:

This chatbot was also integrated with Lambda and SNS to retrieve messages and track occurrences. After creating the lambda function and SNS, I copied the Function Arn from the lambda function to the SNS for me to test the updated chatbot. Once I deployed the code changes using Python in the lambda function, I created an Alias under Amazon Lex to invoke its use with the lambda function as a source. Warning: The chatbot fulfillment needs to check that you're using a lambda function for the chatbot to work or an error will be thrown.



CloudWatch:

You can view the chatbots integration of your lambda function with SNS and Cloudwatch by going to Monitor > Logs > View CloudWatch logs to view the timestamp and message.



The screenshot displays the AWS CloudWatch console interface. The left-hand navigation pane includes sections for 'Alarms', 'Logs', 'Metrics', 'X-Ray traces', 'Events', 'Application monitoring', and 'Insights'. The 'Logs' section is expanded, showing 'Log groups' as the selected option. The main content area shows the breadcrumb path: 'CloudWatch > Log groups > /aws/lambda/PizzaDrinkOrderFunction > 2023/05/10/[\$LATEST]36b8f65713244eea9e40ed37a0c79a48'. Below this, the 'Log events' section is visible, featuring a filter bar with a search input, a 'Clear' button, and time range filters (1m, 30m, 1h, 12h, Custom). A table of log events is displayed with columns for 'Timestamp' and 'Message'. The events include 'INIT_START', 'START', a log message from 'Kelly magheritaseltzer Large', 'END', and 'REPORT'. The interface also includes a 'Display' dropdown and a 'Retry' link for older events.

CloudWatch X

Services Search [Alt+S] N. Virginia

CloudWatch > Log groups > /aws/lambda/PizzaDrinkOrderFunction > 2023/05/10/[\$LATEST]36b8f65713244eea9e40ed37a0c79a48

Log events
You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events Clear 1m 30m 1h 12h Custom Display

Timestamp	Message
No older events at this moment. Retry	
2023-05-10T01:38:04.359-04:00	INIT_START Runtime Version: python:3.10.v2 Runtime Version ARN: arn:aws:lambda:us-east-1::runtime:77
2023-05-10T01:38:04.472-04:00	START RequestId: a2089f22-2b51-4bc3-9cca-93efe6f7e913 Version: \$LATEST
2023-05-10T01:38:04.478-04:00	Kelly magheritaseltzer Large
2023-05-10T01:38:04.479-04:00	Kelly magheritaseltzer Large
2023-05-10T01:38:04.479-04:00	END RequestId: a2089f22-2b51-4bc3-9cca-93efe6f7e913
2023-05-10T01:38:04.479-04:00	REPORT RequestId: a2089f22-2b51-4bc3-9cca-93efe6f7e913 Duration: 7.05 ms Billed Duration: 8 ms Memor
No newer events at this moment. Auto retry paused. Resume	