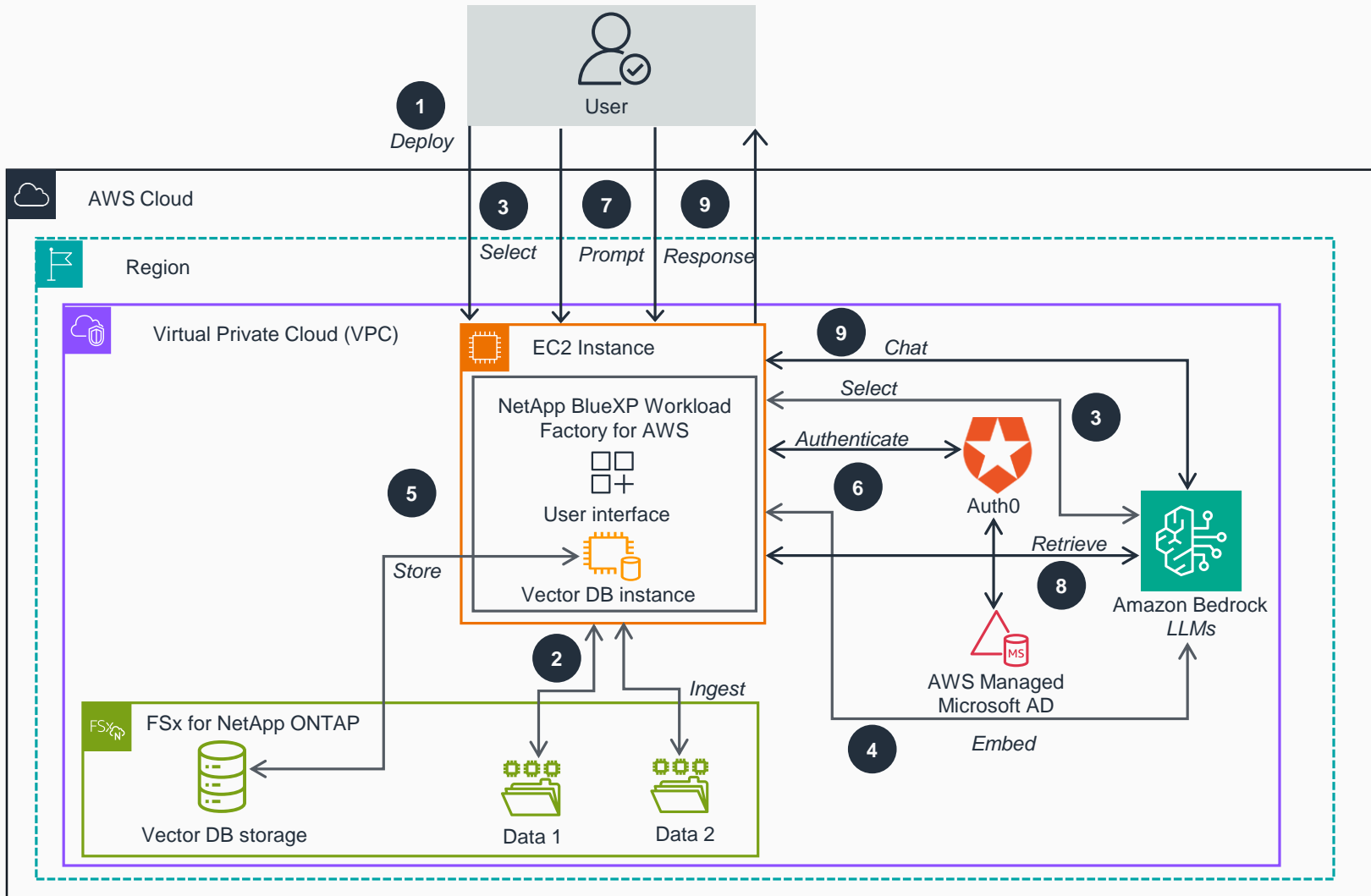


Guidance for Deploying Enterprise Apps with NetApp BlueXP Workload Factory for AWS and Amazon FSx for NetApp ONTAP

This architecture diagram shows how to connect your Amazon FSx for NetApp ONTAP data with Amazon Bedrock to provide your team access to your knowledge base.



- 1 The user deploys the NetApp BlueXP Workload Factory for AWS application into their AWS account, running on an **Amazon Elastic Compute Cloud** (Amazon EC2) instance.
- 2 The NetApp BlueXP Workload Factory for AWS connects to specified source data volumes (Data 1, Data 2) residing in **Amazon FSx for NetApp ONTAP**. This data can originate from on-premises or the AWS Cloud. The NetApp BlueXP Workload Factory for AWS monitors for source data updates, such as new files added, and ingests those data changes.
- 3 The user selects an available Large Language Model (LLM) supported by the NetApp BlueXP Workload Factory for AWS to be used by **Amazon Bedrock** for response generation.
- 4 The NetApp BlueXP Workload Factory for AWS initiates the embedding process using the Amazon Titan in Amazon Bedrock LLM.
- 5 The NetApp BlueXP Workload Factory for AWS Lance DB open source vector database (DB) component runs on **Amazon EC2**. This component stores the vector embeddings received from **Amazon Bedrock** in the vector DB storage within **FSx for ONTAP**.
- 6 The user authenticates through the NetApp BlueXP Workload Factory for AWS application user interface (UI) to Microsoft Active Directory Domain Services using Auth0.
- 7 The user sends a text prompt to the NetApp BlueXP Workload Factory for AWS chatbot application.
- 8 Based on the user's prompt, the NetApp BlueXP Workload Factory for AWS chatbot retrieves Retrieval-Augmented Generation (RAG) vector data from the documents stored in the accessible data volumes (Data1, Data2).
- 9 The NetApp BlueXP Workload Factory for AWS component uses **Amazon Bedrock** to generate a response based on results from the associated LLM combined with RAG. It then displays the response to the user through its user interface.

