IT OPERATIONS SECURITY

SECURING CHATOPS IN AWS

The expertise offered by Storm Reply in the field of governance-focused, audit-friendly, service features with applicative compliance or audit standards, makes the company an ideal partner for the design, development and maintainability of secure-focused enterprise-level multi or hybrid cloud solutions in AWS.

As medium to large-size enterprises expand their IT infrastructure and investments, effectively managing the efforts from a large number of IT experts and their fragmented geographical locations, emphasizes the need for an agile and centralized solution to securely manage IT Operations and communication.

Storm Reply supports companies on their digital transformation journey, focusing on agile solutions ranging from Security Managed Services, Big Data, IoT, and Machine Learning, providing a clear overview to all value chain's stakeholders on the improvements, insights and value added that running a solution in AWS can provide.

6th October, 2020

CHATOPS, WHAT IS IT?

ChatOps is an IT Operational paradigm for which IT Professionals (operators, engineers, developers, and key stakeholders) communicate with each other and interact with the systems they operate by leveraging on a centralized chat solution, usually developed with AI-powered ChatBots. The main advantage of adopting ChatOps is that of speeding up collaboration among relevant and key stakeholders in order to increase time efficiencies and helping companies shorten their time-to-market timelines and properly adopting an agile footprint.

Common operations such as consulting the status of business-critical systems, the creation of Jira tickets, the orchestration of CI/CD Pipelines, responding to incidents, increasing resources to meet a spike in demand, or mitigating DDoS attacks, are all examples of what can be addressed via ChatOps. Cybersecurity is one, if not the most, important aspect of ChatOps as key processes and systems are usually involved. ChatOps is deployed with the support of authentication and authorization layers, end-to-end encryption, logging, and sessions' traceability for the necessary level of governance, future reference or auditing purposes.

COMPONENTS

CHATTING TOOLS. It corresponds with the UI (User Interface) from which users connect and interact among them and with the services integrated in the platform. Chatting tools are either embedded in web or mobile applications, or provided by third parties such as:

- Slack
- Microsoft Teams
- Facebook Messenger
- Telegram
- Hangouts Chat

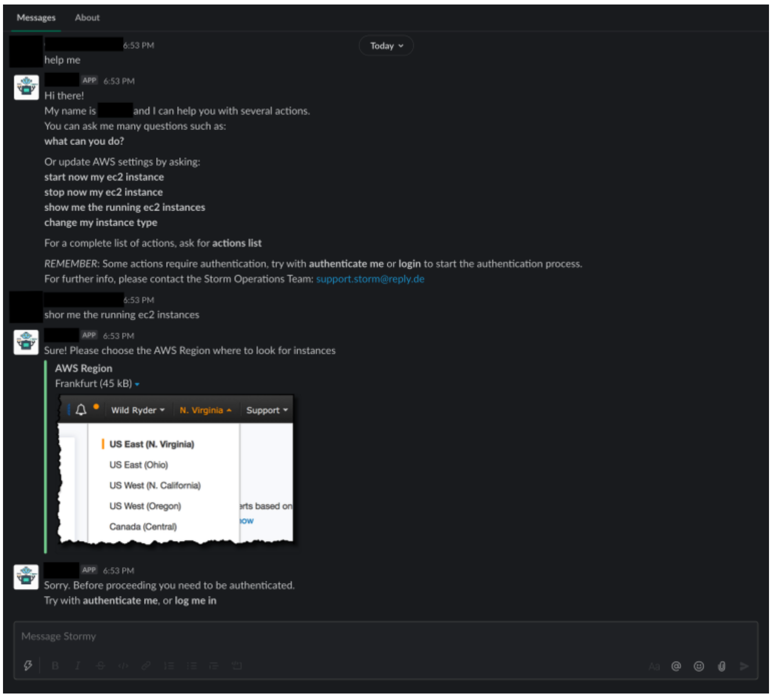
BOTS. The Bot is the "middleware" of the ChatOps paradigm. It offers the necessary level of abstraction and security between the backend services layer and the UI layer for all the requests or APIs involved in the solution and provides developers with the possibility of configuring the necessary "intents" for the IT Operations and communication involved. It is usually powered by Artificial Intelligence (AI) in order to improve its functionality along the time and interactions based on user or systems behaviour, or to perform complex tasks such as software regression testing. The bot server is a hardened gateway to ensure exchange of data and information between the frontend UI or collaboration tool and the backend services.

SERVICES INTEGRATIONS

There is no intrinsic limit on what can be managed or orchestrated with a ChatBot in terms of services integrations. With the latest trend of developing new solutions or extending existing ones in the cloud, several enterprises have adopted a multi-cloud or hybrid-cloud approach in which either different cloud providers or on-premise systems are linked together as a unique ecosystem.

Ad-hoc or built-in services integrations can be deployed to run straightforward transactions with core on-premise or cloud systems, as well as complex, automatic flows that can increase the entire organization’s productivity.
These integrations support the automation and creation of a unified language across teams to accelerate DevOps, SecOps, FinOps, especially between R&D and Operations teams.

A simple example when integrating a ChatBot solution with AWS could be that of consulting the current status of EC2 instances or performing normal DevOps operations such as changing instance type, starting or stopping servers, etc. As required, in order to acquire permissions for calling the necessary APIs, it is necessary for the user to be authenticated.



[redacted] 6:53 PM
help me

[redacted] APP 6:53 PM
Hi there!
My name is [redacted] and I can help you with several actions.
You can ask me many questions such as:
what can you do?

Or update AWS settings by asking:
start now my ec2 instance
stop now my ec2 instance
show me the running ec2 instances
change my instance type

For a complete list of actions, ask for **actions list**

REMEMBER: Some actions require authentication, try with **authenticate me** or **login** to start the authentication process.
For further info, please contact the Storm Operations Team: support.storm@reply.de

[redacted] 6:53 PM
shor me the running ec2 instances

[redacted] APP 6:53 PM
Sure! Please choose the AWS Region where to look for instances
AWS Region
Frankfurt (45 kB)

[redacted] APP 6:53 PM
Sorry. Before proceeding you need to be authenticated.
Try with **authenticate me**, or **log me in**

Figure 1 - AWS Operations with Slack and Amazon Lex

THREATS AND VULNERABILITIES

Depending on the organization’s compliance, auditing policies, sensitivity and security level required by the data and systems being treated, a dedicated assessment and evaluation of threats, vulnerabilities and risks need are performed on a per-solution basis.

Among the most common threats to address there are spoofing, impersonating someone else, data tampering, data theft, repudiation, information disclosure, systems intrusion, DDoS attacks, etc., all addressable with the proper security measures in place.

Vulnerabilities, on the other hand, are defined as ways that a system can be abused or compromised by leveraging bugs that are not properly mitigated, or are not yet identified due to the complexity of the solution, when the system is not well maintained, has poor coding, lacks the proper cybersecurity and governance measures, or due to human errors. Depending on internal organization’s policies, penetration testing, software vulnerability assessments and cloud-security related assessments could be performed regularly so to mitigate the risks derived.

SECURITY FRAMEWORK BY STORM REPLY

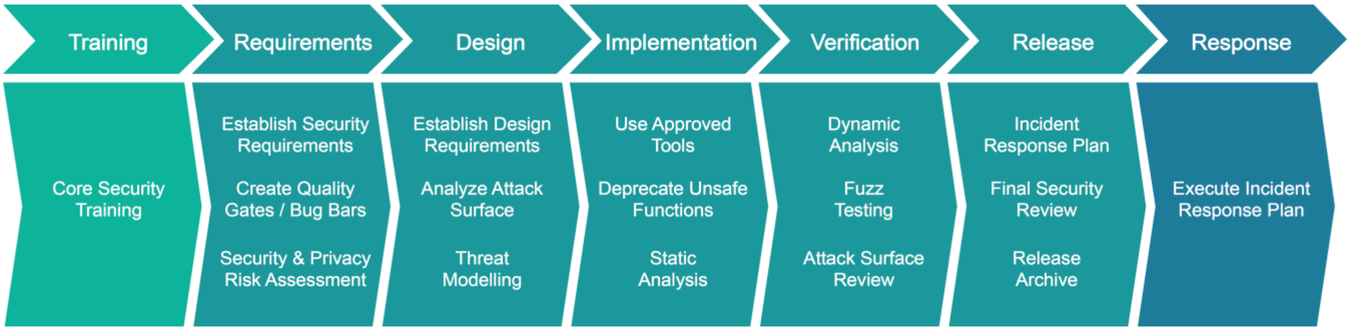
Storm Reply follows the [AWS Cloud Adoption Framework]() and the [AWS Well Architected Framework]() with its design principles and architectural best practices pillars such as Security, Reliability, Performance Efficiency, Cost

Optimization and Operational excellence.

Any security related solution or project takes into account key strategic aspects when deploying solutions in the cloud such as Prevent, Detect, Respond, and Remediate.

The following framework has been adapted to match a comprehensive security posture for a ChatOps Solution leveraging on current AWS security services.

DEFINE A SECURITY DEVELOPMENT LIFECYCLE. As part of the agile methodology applied in infrastructure deployment and application development, it is also important to define a clear and comprehensive development cycle focused on including cybersecurity in every step of the project plan.



Training	Requirements	Design	Implementation	Verification	Release	Response
Core Security Training	Establish Security Requirements Create Quality Gates / Bug Bars Security & Privacy Risk Assessment	Establish Design Requirements Analyze Attack Surface Threat Modelling	Use Approved Tools Deprecate Unsafe Functions Static Analysis	Dynamic Analysis Fuzz Testing Attack Surface Review	Incident Response Plan Final Security Review Release Archive	Execute Incident Response Plan

The different aspects should be clear and managed with a bottom-up approach, collecting feedback and enforcing best practices adoption or acquisition down from Security-Operations teams, up to Development and Management teams. In order to ensure business goals are successfully achieved, security needs to be a priority, not an option.

CATEGORIZE ATTACKS AND DEFINE MITIGATION TECHNIQUES. As part of the design process, it is fundamentally important to properly identify, categorize and mitigate both threats and vulnerabilities on the specific solution.

Category	Attack Intentions	Mitigation Techniques
Spoofing	Illegally access and use another user's credentials. Impersonating something or someone else’s account.	▪ Appropriate authentication (strong password) ▪ Protect sensitive data ▪ Don't store secrets in unsafe storage
Tampering	Aimed to maliciously change/modify data	▪ Hashes ▪ Digital signatures ▪ Tamper-resistant protocols ▪ Append-only audit logging
Repudiation	Aimed to perform illegal operations in a system	▪ Digital signatures ▪ Audit trails ▪ Use NTP and log timestamps
Information Disclosure	Data theft	▪ Encryption ▪ Privacy-enhanced protocols ▪ Exclude sensitive data in logs (e.g. GDPR)
Denail of Service	Aimed to deny access to valid users	▪ Appropriate authentication/authorization ▪ Filtering and Throttling ▪ QoS
Elevation of Privilege	Aimed to gain privileged access	▪ Least privilege design principle ▪ Strong IAM

ADJUST SECURITY MEASURES FOR SMOOTH USER EXPERIENCE. Username & Password or biometrics like fingerprint and face recognition are well known authentication methods for desktop, web and mobile apps. Conversational apps may require a more natural to the new medium approach in order to ensure adoptability while still guaranteeing the required level of security.

Depending on the actions allowed and the usability to be provided to users, customized sessions' duration need to be configured. For this task, AWS makes it easy for developers as its AWS Security Token Service (STS) service allows temporary assumption of Roles defined with policies and permissions. In case other federation services might be involved (e.g. AWS Managed Microsoft AD with ADFS, Amazon Cognito, etc.), the overall end-to-end session duration will need to be configured for a seamless and secure user experience.

SECURE YOUR PUBLIC ENTRY-POINTS WITH TLS/SSL AND WAF. With AWS WAF it is possible to protect the different APIs exposed to the internet by defining either AWS-managed rules or customized rules aiming to better match the application and the security levels needs. In terms of detection and mitigation of DDoS attacks, the out-of-the-box availability of the AWS Shield service makes sure the application faces minimized downtimes without engaging the support from AWS.

As far as the treatment of data in-transit is concerned, SSL certificates and TLS configuration are enforced. Even when dealing with internal communication, it is best practice to make sure communication is always encrypted with the supported TLS protocols. Amazon makes sure to expose its services through encrypted TLS communications and enforcing Signature Version 4 so that every interaction with its APIs and services covers authentication, authorization and encryption.

APPLY CONSISTENT AUTHENTICATION & AUTHORIZATION (MFA, DA). These two key features of Identity and Access Management (IAM) can be utilized with the usage of different architectures and AWS Security Services, based on the UI or chatting tool being used and the security level required by the ChatOps platform. AWS makes it easy to integrate an IdP of choice for authentication and authorization, without integrating with LDAP, or Active Directory directly.

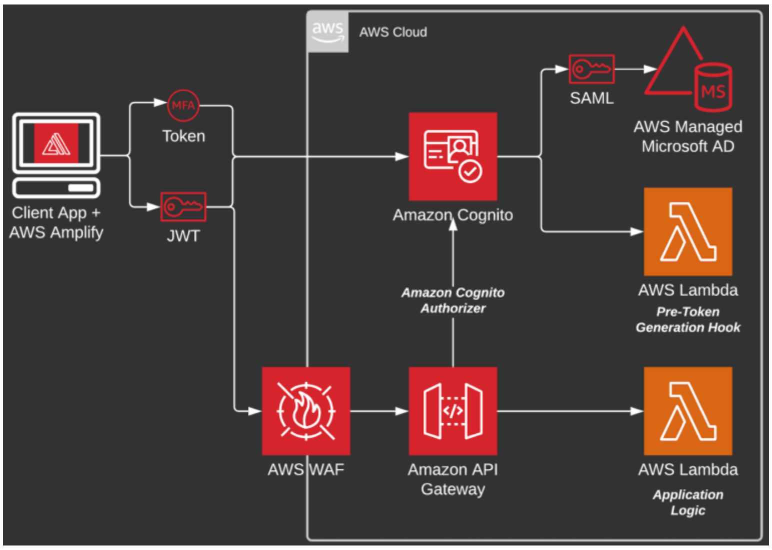


Figure 2 - ChatOps Authentication and Authorization Layer

As depicted in the serverless architecture below, by using Amazon Cognito, both the MFA strategy and the structure of the JWT token can be normalized, so that adding multiple IdPs, social login providers, and even regular username and password-based users (stored in user pools) for authentication can be managed without changing a single line of code.Amazon API Gateway's native integration with Amazon Cognito's user pools authorizer streamlines the validation of the JWT integrity, which then allows the backend to make the necessary authorization decisions for the solution.

ENSURE CUSTOMERS DATA PRIVACY. With the usage of AWS Key Management Service (KMS) it is possible to ensure and enforce a proper level of governance and security for the access and treatment of data at-rest. In particular, for what concerns sessions-related logs, access logs, end-to-end logs, and WAF logs, it is possible to configure a dedicated KMS key for data logged or stored in Amazon CloudWatch Logs or Amazon S3. Amazon takes care of the necessary services integration in order to ease the adoption and configuration of encryption at-rest among its services.

CONSTANTLY ENFORCE ENTERPRISE REQUIREMENTS FOR COMPLIANCE.

- Check for client's specific Cybersecurity requirements and compliance programs. Whether it is a PCI, GDPR, SOC or HIPAA, make sure that any aspect of the solution is compliant and considered before and during the development phase.
- Ensure service redundancy, and be prepared for any situation in case something breaks (Integrated SIEM for service and infrastructure availability).
- Evaluate periodic Risk Assessments for the overall solution.
- Perform periodic functional/penetration tests: the often you check your product, the sooner you find out its strengths and possible weaknesses.

STORM REPLY

Storm Reply provides extensive and comprehensive specialist knowledge about AWS Cloud Cybersecurity at the Corporate or Enterprise level, and in several areas, thanks to other specialized business units such as Spike Reply and Communication Valley Reply.