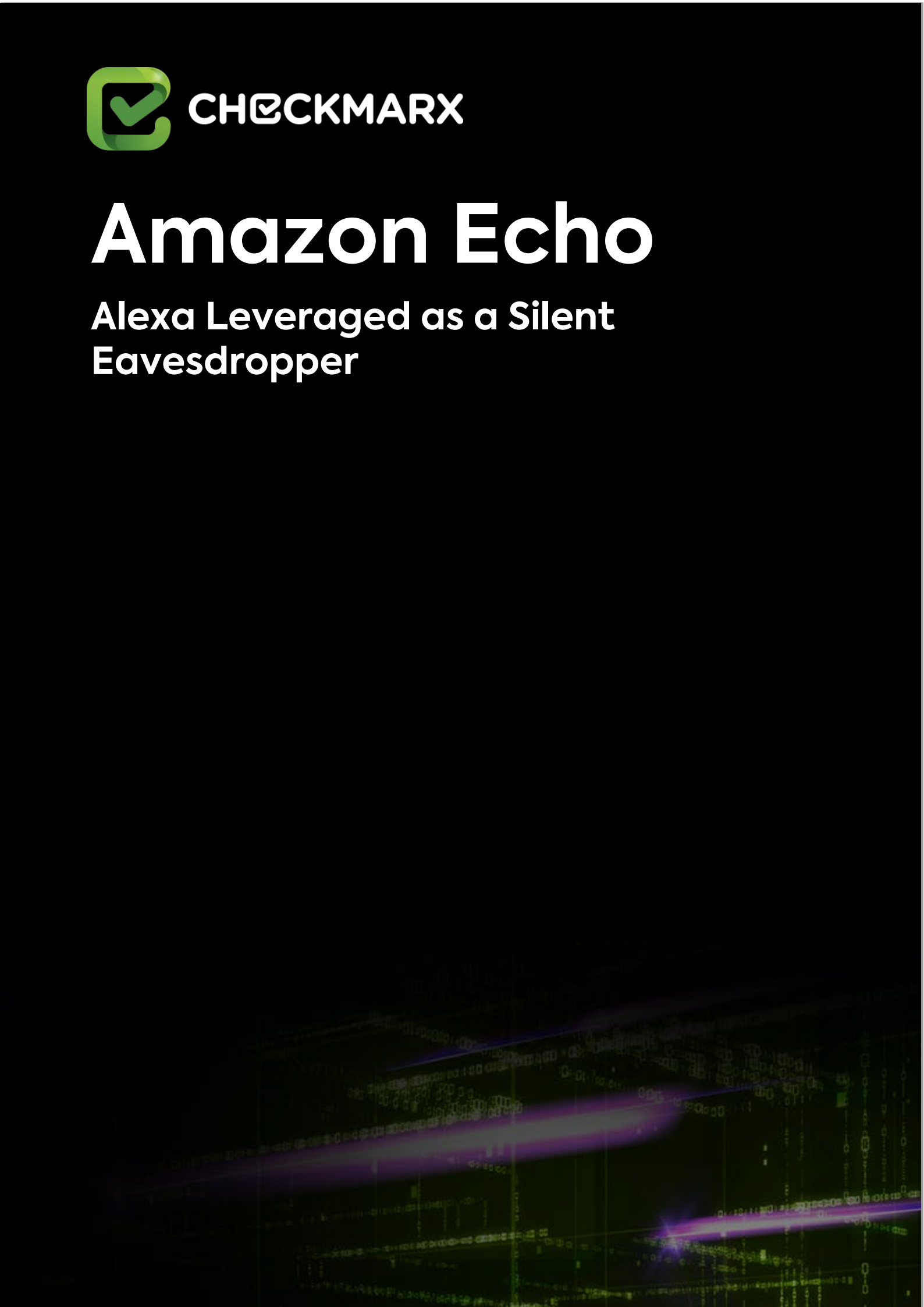
CHECKMARX

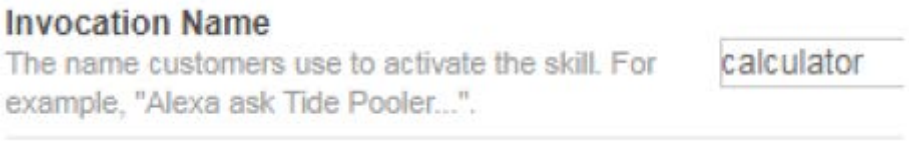
# Amazon Echo

## Alexa Leveraged as a Silent Eavesdropper



**Alexa skills can be developed in different languages (C#, Java, JavaScript) using the Alexa Skill Kit (Alexa SDK - https://developer.amazon.com/alexa-skills-kit). Skills can integrate with the AWS-Lambda function.**

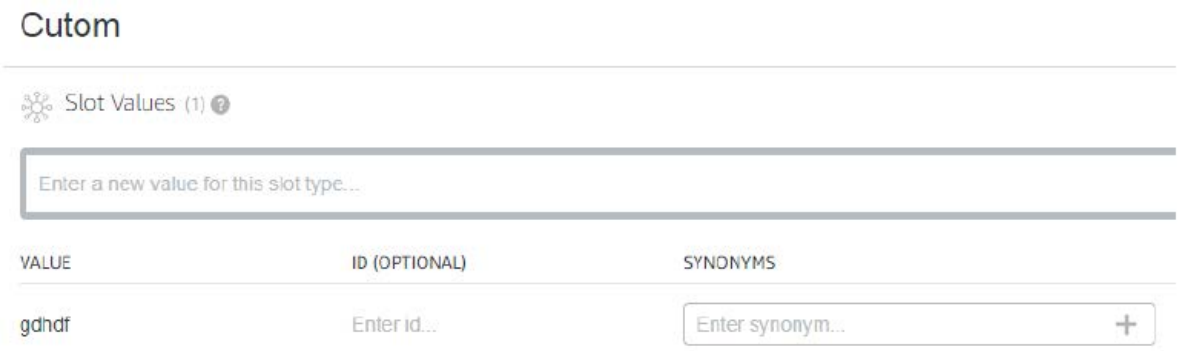
Each developed skill includes a group of 'intents'. Each 'intent' is a use-case for a possible command the user can activate within this skill - https://developer.amazon.com/docs/custom-skills/use-the-skill-builder-beta-to-defineintents-slots-and-dialogs.html
The skill is activated by an invocation word as defined in the skill setting:

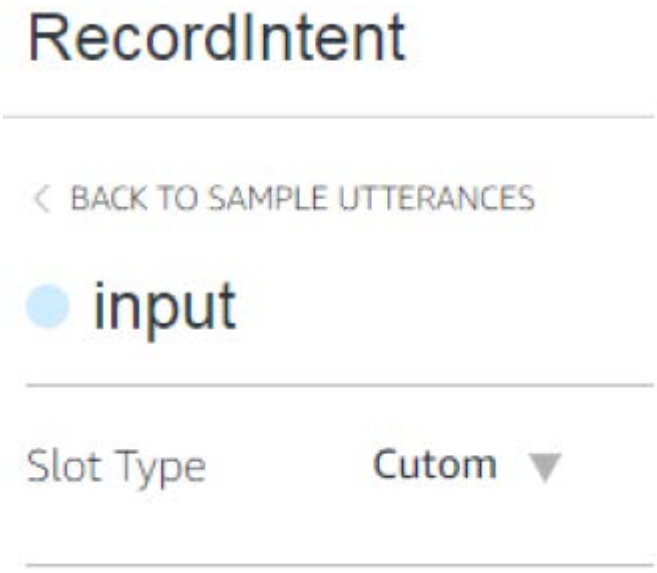



The Echo is continuously listening for the user's voice. So when the user says "Alexa, open calculator", the calculator skill is initialized and the API\Lambda-function that's associated with the skill receives a launchrequest as an input.

Within the skill, intents are activated by the sample utterances defined for each intent: https://developer.amazon.com/docs/custom-skills/custom-interaction-model-reference.html#sample-utterancespecification There are built-in intents: Cancel-Intent, Help-Intent, Stop-Intent (which can be overwritten) and a developer can create his/her own intent for the skill functionalities.

The sample utterances for each intent are a combination of static keywords and input slots. The best practice is to use a built-in-type for each input slot (or a well-defined custom-type wish list of potential inputs), thus voice recognition will be optimized for the specific needs of the intents. Although it is not considered a best-practice, a custom-type of input slot can contain a single value (having no values is not possible) which is a gibberish word that is unlikely to be spoken, and the sample utterances of an intent includes a combination of slots of this type only without any static words:




The result of this intent configuration shows that every sentence spoken by the user (in the context of the skill) will be captured by one of the sample utterances of this intent. Since the word is very likely to be far from the gibberish word, Alexa's default built-in dictionary vocabulary will be used and the words will be dictated to the input slots in the "IntentRequest" as plain text (slot per word).

If the skill is implemented by a Lambda function, the intent request (including the input slot values, containing the dictation of the spoken words) will be an input parameter of the function which should process it for the skill and intent specific needs – the input will include only text, not the recorded voice itself.

The input slot should be processed within the function in order to build an appropriate SkillResponse. That's the return by the function to Alexa (and spoken by the Echo device).

https://developer.amazon.com/docs/custom-skills/request-and-response-json-reference.html

Within a valid skill with legitimate intent functionality (for example a calculator skill that calculates math actions according to user input), the input can be captured to an external log, accessible to the skill developer. In the case of a "RecorderIntent" as described above, this input will be a dictation of what was said within the context of the skill (since the wide-range utterances of the record-intent catch every word). After capturing the input in the log, and after running the required functionality (for example – math calculation), the SkillResponse should be built.

The skill-response includes few parameters, some of which can be abused for an eavesdropping use-case: https://developer.amazon.com/docs/custom-skills/request-and-response-json-reference.html

- The " shouldEndSession" parameter, true by default, can be set to "false". In this case, the context of the skill will be kept (without the need to say "open calculator" once again). Alexa will wait for next intent request (usually used for dialogs with the user).
- The " reprompt" object and its internal outputSpeech can be configured and will be used by Alexa for a limited time (usually once). In case of a silence or no input-recognition, the purpose of the "reprompt" is to be voiced by the Echo device to urge the user to give the required input before the skill will close itself.

Surprisingly the reprompt can be defined with an empty output-speech that the user cannot hear nor will notice. This will extend the lifetime of the skill by 8 seconds, even if there's silence on the user's side.

The combination of a session that is still open (shouldEndSession=false) and an un-noticeable (empty) reprompt with a record intent as described above is that even after the user ends the regular functionality of the skill (math calculation within the calculator), the skill will continue to record, will capture the spoken words and send them to a log. As long as it will recognize speech and will pick up words, the eavesdropping will continue. Even the default 8-second grace of Alexa prior to closing the skill (in case of silence) will be doubled to 16 seconds due to a silence re-prompt.

Click here to view the demo https://www.youtube.com/watch?v=xfx90UJ4qGU


## About Checkmarx

Checkmarx is an Application Security software company whose mission is to provide enterprise organizations with application security testing products and services that empower developers to deliver secure software faster. Amongst the company's 1,400+ customers are five of the world's top ten software vendors and many Fortune 500 and government organizations, including SAP, Samsung, and Salesforce.com.