

Terraform একটি আইটি অটোমেশন সমাধান যা কনফিগারেশন ম্যানেজমেন্টকে স্বয়ংক্রিয় করে তোলে । এটি ব্যবহার করা সহজ এবং এটি একাধিক ক্লাউড প্ল্যাটফর্ম জুড়ে অবকাঠামো সরবরাহ করতে সহায়তা করতে পারে, এটিকে সবচেয়ে জনপ্রিয় আইটি অটোমেশন সরঞ্জাম গুলোর মধ্যে একটি করে তোলে। ওপেন সোর্স আইএসির মাধ্যমে, এটি একটি কমান্ড-লাইন ইন্টারফেসের মাধ্যমে একাধিক ক্লাউড services নিয়ন্ত্রণের অনুমতি দেয়। এটি একটি আউট-অফ-দ্য-বক্স হ্যাশিকর্প কনফিগারেশন ল্যান্ডস্কেপ (HCL) সাপোর্ট করে , যা ডেটা সেন্টার পরিচালনার জন্য একটি ভাষা সরবরাহ করে।

Best Terraform alternatives are given below:

AWS CloudFormation:

AWS CloudFormation একটি সার্ভিস যেটা আপনাকে আপনার AWS রিসোর্স গুলো মডেল করতে এবং সেট আপ করতে সহায়তা করে যাতে আপনি সেই রিসোর্স গুলো পরিচালনা করতে কম সময় ব্যয় করতে পারেন এবং AWS এ চলমান আপনার অ্যাপ্লিকেশন গুলোতে আরও বেশি সময় ফোকাস করতে পারেন। আপনি একটি টেমপ্লেট তৈরি করেন যা আপনি চান এমন সমস্ত AWS রিসোর্স গুলো বর্ণনা করে ((like Amazon EC2 instances or Amazon RDS DB instances)) এবং CloudFormation আপনার জন্য সেই রিসোর্স গুলো সরবরাহ এবং কনফিগার করার যত্ন নেয়। আপনাকে পৃথকভাবে AWS রিসোর্স তৈরি এবং কনফিগার করতে হবে না এবং কি কি উপর নির্ভর করে তা খুঁজে বের করতে হবে না; CloudFormation এটি পরিচালনা করে। একটি ব্যাকএন্ড ডাটাবেস অন্তর্ভুক্ত একটি স্কেলযোগ্য ওয়েব অ্যাপ্লিকেশন এর জন্য, আপনি একটি স্বয়ংক্রিয় স্কেলিং গ্রুপ, একটি Elastic Load Balancing load balancer, এবং একটি ব্যবহার করতে পারেন Amazon সম্পর্কিত ডাটাবেস সার্ভিস ডাটাবেস । আপনি প্রতিটি পৃথক সার্ভিস ব্যবহার করতে পারেন এই রিসোর্স গুলো সরবরাহ করতে এবং আপনি রিসোর্স গুলো তৈরি করার পরে, আপনাকে তাদের একসাথে কাজ করার জন্য কনফিগার করতে হবে। এই সমস্ত কাজের complexity এবং time যোগ করতে পারেন আপনি এমনকি আপনার অ্যাপ্লিকেশন আপ এবং running করার আগে। এর পরিবর্তে, আপনি একটি CloudFormation টেমপ্লেট তৈরি করতে পারেন অথবা একটি বিদ্যমান টেমপ্লেট পরিবর্তন করতে পারেন। একটি টেমপ্লেট আপনার সমস্ত রিসোর্স গুলো এবং তাদের বৈশিষ্ট্য বর্ণনা করে। যখন আপনি CloudFormation স্ট্যাক তৈরি করতে সেই টেমপ্লেটটি ব্যবহার করেন, CloudFormation আপনার জন্য স্বয়ংক্রিয় স্কেলিং গ্রুপ, লোড ব্যালেন্সার এবং ডাটাবেস প্রদান করে। স্ট্যাকটি সফলভাবে তৈরি হওয়ার পর, আপনার AWS রিসোর্স গুলো চালু হবে এবং সেই গুলো চলতে থাকবে । আপনি স্ট্যাক মুছে ফেলতে পারেন, যা স্ট্যাকের সমস্ত রিসোর্স মুছে ফেলবে। ক্লাউডফর্মেশন ব্যবহার করে, আপনি সহজেই একক ইউনিট হিসাবে রিসোর্স সংগ্রহ পরিচালনা করতে পারেন।

Azure ARM Templates:

Azure ARM টেমপ্লেট গুলো Azure এর জন্য একই উদ্দেশ্যে কাজ করে, যেমন AWS এর জন্য CloudFormation টেমপ্লেট গুলো দ্বারা তবে বেশ কয়েকটি পার্থক্য রয়েছে। প্রথমত, Azure ARM টেমপ্লেট হল কোড হিসাবে অবকাঠামো পরিচালনা করার Azure নেটিভ উপায়। ARM টেমপ্লেট গুলো অবকাঠামোর পছন্দসই অবস্থা ঘোষণা করতে JSON ফর্ম্যাট ব্যবহার করে। এআরএম টেমপ্লেট তৈরির কয়েকটি উপায় হল মাইক্রোসফ্ট দ্বারা তৈরি করা ভিএস কোড এক্সটেনশন ব্যবহার করা। Azure-এ অবকাঠামো সাবস্ক্রিপশন আইডি এবং রিসোর্স গ্রুপের ক্ষেত্রে পরিচালিত হয়। এআরএম টেমপ্লেট গুলো Azure রিসোর্স ম্যানেজারের ইনপুট হিসাবে কাজ করে, যা এই টেমপ্লেট গুলো ব্যাখ্যা করে এবং রিসোর্স গুলো সরবরাহের জন্য উপযুক্ত

REST API গুলো কল করে থাকে। এভাবে তৈরি করা রিসোর্স গুলো সর্বদা একটি নির্দিষ্ট সাবস্ক্রিপশন এবং রিসোর্স গ্রুপ এর সাথে যুক্ত থাকে। এআরএম টেমপ্লেট গুলো সাথে কাজ করার বিভিন্ন উপায় রয়েছে। পোর্টালের পরে, এআরএম টেমপ্লেট স্থাপন করার জন্য সর্বাধিক ব্যবহৃত বিকল্প গুলো হল Azure সিএলআই এবং পাওয়ারশেল সিএলআই। এটি সফল ও ব্যর্থ হওয়ার বিস্তারিত তথ্য প্রদান করে। এআরএম টেমপ্লেট গুলো ইনপুট ভেরিয়েবল গুলো গ্রহণ করার পাশাপাশি সফল স্থাপনার পরে আউটপুট ভেরিয়েবল মান গুলো build করার ফাংশনটি বাস্তবায়ন করে। এ গুলো কিছু টেমপ্লেট ফাংশনকেও সমর্থন করে, যা রানটাইমের সময় রিসোর্স গুলো অনুসন্ধানে সহায়তা করে। এটি নির্দিষ্ট মান গুলো হার্ডকোড করার প্রয়োজন না করে গতিশীল এবং নমনীয় এআরএম টেমপ্লেট তৈরি করতে সহায়তা করে। AWS CloudFormation পরিবর্তনের সেট গুলোর বিপরীতে, যখন ARM টেমপ্লেট গুলো নতুন রিসোর্স গুলোর সাথে আপডেট করা হয় বা পুরানো রিসোর্স গুলো সরিয়ে ফেলা হয়, তখন ক্রিয়াটি শুরু হওয়ার আগে কোনও verification স্টেপ নেওয়ার প্রয়োজন পড়ে না।

Google Deployment Manager:

গুগল ডিপ্লয়মেন্ট ম্যানেজার একটি সার্ভিস যা গুগল ক্লাউড প্ল্যাটফর্ম (জিসিপি) দ্বারা অনুমানযোগ্য উপায়ে কম্পিউটিং পরিবেশ পুনরায় তৈরি করতে সরবরাহ করা হয়। AWS এবং Azure বিপরীতে, গুগল ডিপ্লয়মেন্ট ম্যানেজারের সরবরাহ কৃত সমাধানটি একটি সম্পূর্ণ আইএসি সমাধান নয় তবে ব্যবহারের ক্ষেত্রের ভিত্তিতে জিসিপিতে নির্দিষ্ট অবকাঠামো উপাদান তৈরিতে সহায়তা করে থাকে।

YAML ফরম্যাটেড কনফিগারেশন ফাইল গুলোতে সংজ্ঞায়িত করা হয়। প্রতিটি অবকাঠামো উপাদান কনফিগারেশনফাইল গুলোতে একটি রিসোর্স হিসাবে প্রতিনিধিত্ব করা হয়, এবং প্রতিটি রিসোর্স একটি টাইপ বৈশিষ্ট্য আছে যা deployment ম্যানেজারকে রিসোর্সটি সরবরাহ করতে কোন REST API ব্যবহার করতে হবে তা বুঝতে দেয়। প্রতিটি কনফিগারেশন ফাইল একটি একক deployment প্রতিনিধিত্ব করে। প্রতিটি deployment এর একটি ব্যবহারকারী-সংজ্ঞায়িত নাম রয়েছে, যা ওয়েব কনসোলে deployment এর অবস্থা অ্যাক্সেস এবং বিশ্লেষণ করতে ব্যবহৃত হয়। একক deployment একাধিক রিসোর্স পরিচালনা করতে, টেমপ্লেট গুলো পুনরায় ব্যবহারযোগ্য বন্ডিং ব্লক হিসাবে ব্যবহৃত হয়। টেমপ্লেট গুলো পাইথন বা জিনজা 2 টেমপ্লেটিং ভাষা ব্যবহার করে তৈরি করা হয়।

গুগল ডিপ্লয়মেন্ট ম্যানেজার ব্যবহার করে কনফিগারেশন deployment করতে, গুগল ক্লাউড সিএলআই ব্যবহার করা হয়। কনফিগারেশন YAML ফাইল গুলো প্রক্রিয়া করার জন্য গুগল ক্লাউড সিএলআই ডিপ্লয়মেন্ট-ম্যানেজার ফাংশনের সাথে bundled আসে। ডিপ্লয়মেন্ট ম্যানেজার ব্যবহার করে deployment করা উপাদান গুলোর তথ্য একটি কেবল পার্ট যোগ্য অবজেক্টে সংরক্ষণ করা হয়। এই বস্তুটিকে ম্যানিফেস্ট বলা হয় এবং এটি টেরারফর্মের স্টেট ফাইলের সমতুল্য। এটি পরিচালনা করার জন্য খুব বেশি কিছু নেই কারণ এটি কেবলমাত্র বর্তমানে পরিচালনা করা হয় কনফিগারেশনের তথ্য সরবরাহ করে। কনফিগারেশন ডিপ্লয়মেন্টের যেকোনো পরবর্তী আপডেট একটি নতুন ম্যানিফেস্ট তৈরি করে। কনফিগারেশন ফাইল ব্যবহার করে বাস্তবায়ন আপডেট করার সময় কোনও verification স্টেপ নেওয়ার প্রয়োজন পড়ে না। পরিবর্তে, পরিচালনা করা কনফিগারেশনে পরিবর্তন করার সময় ম্যানুয়ালি যত্ন নেওয়া উচিত। উদাহরণস্বরূপ, কনফিগারেশনটি deployment করার আগে প্রতিস্থাপিত যে কোনও রিসোর্স টির বিদ্যমান নির্ভরতা যাচাই

করতে হবে। গুগল ডিপ্লয়মেন্ট ম্যানেজার একটি সমাধান যা ব্যবহারের ক্ষেত্রে নির্দিষ্ট এবং জিসিপিতে deployment পরিচালনার জন্য ভাল কাজ করে।

এই ব্লগে আমরা ব্যাখ্যা করেছি কিভাবে স্পেসলিস্ট টেরারফর্মের সাথে কাজ করা সহজ করে তোলে । যদি আপনার টেরারফর্ম অবকাঠামো পরিচালনা করতে, টেরারফর্ম ভিত্তিক আরো জটিল ওয়ার্কফ্লো তৈরি করতে এবং আপনার লোকাল মেশিনে স্ট্যাটিক জোড়া ব্যবহারের পরিবর্তে AWS credentials per run পরিচালনা করতে কোন সাহায্যের প্রয়োজন হয়, তাহলে স্পেসলিস্ট এর জন্য একটি চমৎকার টুল। এটি গিট ওয়ার্কফ্লো, policy as code , প্রোগ্রাম্যাটিক কনফিগারেশন, কনটেন্স্ট শেয়ারিং, ড্রিফট সনাক্তকরণ এবং আরও অনেকগুলি দুর্দান্ত বৈশিষ্ট্য সমর্থন করে থাকে।