# Deploying a Golden AMI Pipeline in AWS with Ansible

Organizations of all sizes and types continue to focus on eliminating the need for repetitive tasks and improving processes. Today, many organizations still rely on using high valued resources to perform manual tasks. Not only is this a waste of time and money, but its highly inefficient and is prone to errors. As an example, there is no need to have a virtual machine (VM) provisioning playbook consisting of multiple pages that someone must read through to deploy a VM. The old saying goes, if you are doing something more than a couple of times, automate it. You may spend more time figuring out how to automate it in the beginning, but it will pay off tremendously each time moving forward.

How does your organization ensure that when a new VM is provisioned, that it's being done efficiently, in a consistent manner, free from errors, with no vulnerabilities? Allowing end users to just spin up VMs of their choosing with no checks and balances is asking for trouble. You may ask yourself, isn't that why we build self-provisioning capabilities into our processes? Yes, but it must be done the right way. Having a golden image pipeline can ensure a standardized process with controls while allowing self-service provisioning. Let's list some of the key concepts of the pipeline.

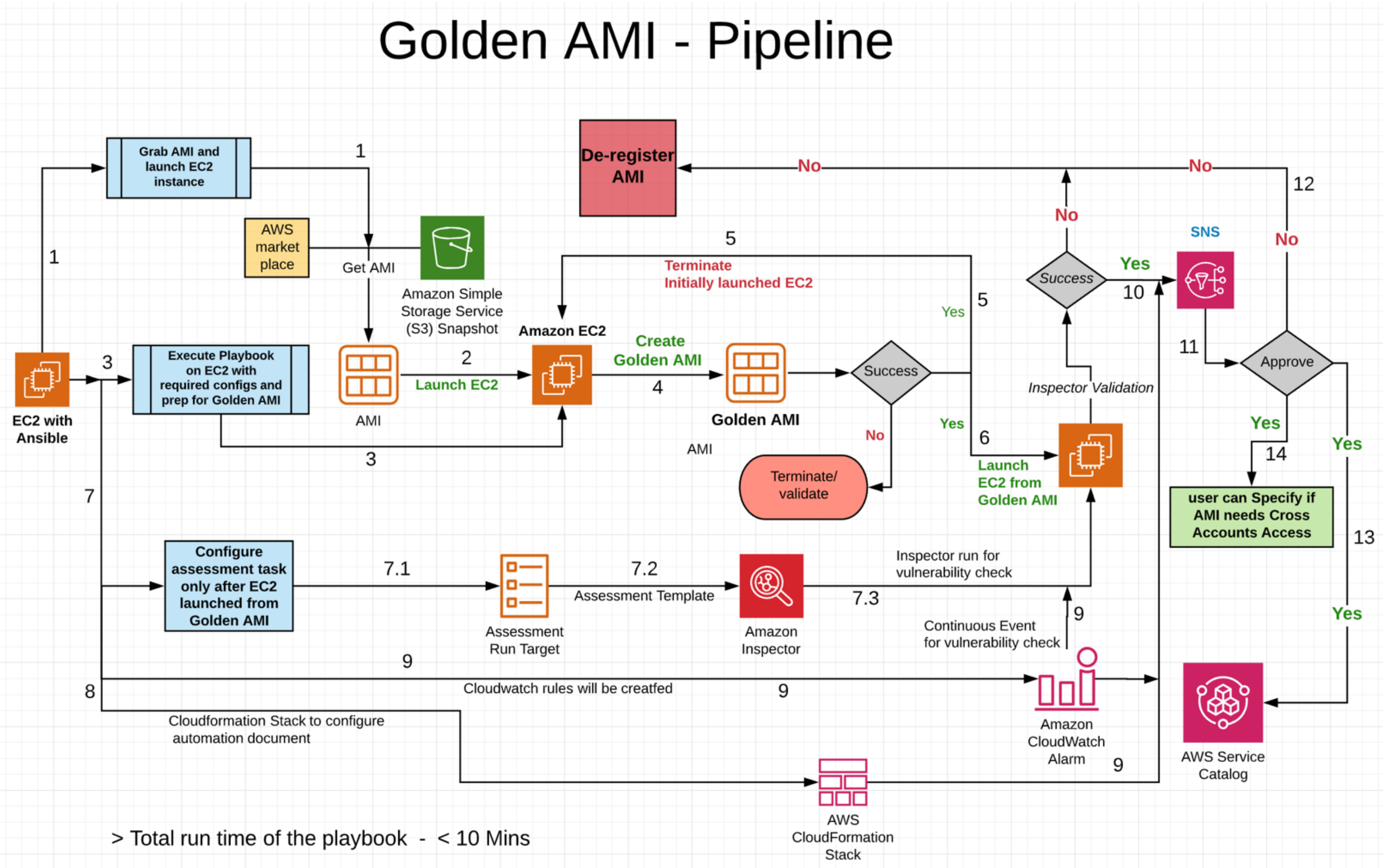
- **Golden Image** - is a standardized template for a VM that has been built exactly the way that you want. It is standardized through configuration, consistent security patching and hardening. It can also include agents for logging, security, performance monitoring and tools necessary to perform various tasks. The use of a golden image saves time and ensures that you have a consistent image that is approved for use by the organization. Collections of golden images may be referred to as a golden repository, gold catalog or golden image library.
- **Hardening** - the operating system (OS) is a key step in building a golden image template. This involves configuring the OS securely, removing unnecessary applications and services, updating and creating rules and policies that govern its use. This is done to minimize exposure to threats by reducing the attack surface and to mitigate risk.
- **Amazon Machine Image (AMI)** - is a template that contains a software configuration and information required to launch an instance in AWS. This can be a private AMI, Amazon-owned AMI or AWS marketplace AMI as the source AMI.
- **Ansible** - is an open source configuration management and automation platform. The tool pushes application code, programs and IT infrastructure setup instructions via modules to managed nodes, whether physical servers, virtual machines (VMs) or cloud instances. It is agentless, meaning that it does not install software on the nodes that it manages only required on one node from where it will be executing. This removes a potential point of failure and security vulnerability and simultaneously saves system resources.
- **Amazon S3 -** is object storage allowing customers to put data in the cloud. Object storage typically includes the data itself, a variable amount of metadata, and a unique identifier that allows applications to read it. S3 also includes very granular security permissions to be able to determine who can access data. This is a very important part of the public cloud and being able to secure applications that can read data can minimize data breaches and leaks.

- **Amazon Inspector -** is an automated security assessment scanner which can evaluate security and possible Vulnerabilities and provide best practice for applications hosted on AWS. AWS Inspector communicates with EC2 instances with the help of agents installed on it and generates assessment findings.
- **Amazon simple Notification Service -** is a fully managed that facilitates message delivery using a publish/subscribe model. SNS supports conveyances, such as HTTP/S, SMS and email, and can deliver push messages to multiple recipients. SNS is often used to push messages directly to supported services.
- **AWS Service Catalog –** allow organizations to create and centrally manage commonly deployed IT services. These are approved products which will allow users to deploy their applications from existing products with ease.
- **CloudFormation –** provides an easy way to create and manage a collection of related AWS resources while provisioning, updating and terminating those resources in an orderly and predictable fashion.
- **Systems Manager (SSM) –** is an automation that makes common deployment tasks of EC2 instance and other AWS resources. Can create custom Documents or use Documents that were pre-defined by AWS. Monitor automation progress and helps to provide detailed information. Allow users to approve or deny a task that will continue or abort the execution**.**

## Golden AMI Pipeline

The golden AMI pipeline enables the creation, patching, inspection, approval, distribution and decommissioning of the golden AMI. The workflow will launch an EC2 instance which has the Inspector agent installed and patch/configure the OS using an ansible playbook. Amazon Inspector will perform a scan of the EC2 instance. The scan will check for Network Reachability, Common Vulnerabilities and Exposures, Center for Internet Security (CIS) Benchmarks, Security Best Practices for Amazon Inspector and Runtime Behavior Analysis. A CloudFormation stack will be created with SNS topic and SSM automation documents. Once the scan has been completed an email will be sent out to a specified user using the SNS topic related to the golden AMI. This email contain links to approve or deny the golden AMI build and a link to Amazon Inspector findings to include possible threats specifying each finding with risk level. After evaluating all the findings, you can proceed to approve or deny the build. If user approves the golden AMI build, then he/she can choose whether to add it to AWS Service Catalog or not (in case organization doesn’t want to use service catalog). If added to service catalog, it will be made available for use to deploy EC2 instances. AMIs can be copied over to multiple regions from service catalog. If not added, then the golden AMI will be updated with tags as Approved-AMI. If user denies the golden AMI build upon reviewing the findings, the AMI will be deregistered and decommissioned.

This below diagram illustrates the workflow of the Golden AMI Pipeline.


Golden AMI - Pipeline
Grab AMI and launch EC2 instance
1
De-register AMI
No
No
12
1
AWS market place
Get AMI
Amazon Simple Storage Service (S3) Snapshot
5
Terminate Initially launched EC2
No
Success
Yes
10
SNS
No
Yes
5
Amazon EC2
EC2 with Ansible
3
Execute Playbook on EC2 with required configs and prep for Golden AMI
AMI
2
Launch EC2
Create Golden AMI
4
Golden AMI
Success
Inspector Validation
11
Approve
3
AMI
No
Yes
6
Yes
Yes
14
Terminate/ validate
Launch EC2 from Golden AMI
user can Specify if AMI needs Cross Accounts Access
7
13
Configure assessment task only after EC2 launched from Golden AMI
7.1
Assessment Run Target
7.2
Assessment Template
Amazon Inspector
Inspector run for vulnerability check
7.3
Continuous Event for vulnerability check
9
Yes
9
8
Cloudwatch rules will be creatfed
9
Amazon CloudWatch Alarm
9
AWS Service Catalog
Cloudformation Stack to configure automation document
AWS CloudFormation Stack
> Total run time of the playbook - < 10 Mins

Golden AMI Pipeline Workflow

1. Use an AMI (either from AWS marketplace or customized AMI which needs to be updated in the ansible variable file)
2. Launch EC2 instance using previously registered AMI
3. Patch, add/modify configs of EC2 instance as per user requirement. Inspector agent will be installed
4. EC2 instance will be used to create an AMI called Golden AMI
    - If the creation of golden AMI failed, process will be halted for troubleshooting
5. If the creation of golden AMI succeeds, then initial instance will be terminated.
6. With the termination of initial instance, a new EC2 instance from golden AMI will be launched.
7. Inspector Assessment target will be created

    7.1 Continuous Inspector Assessment target will be configured for scanning any potential vulnerabilities

    7.2 Amazon Inspector template will be created

7.3 Inspector Assessment run will be triggered for the first time after the golden AMI creation.

8. CloudFormation stack will be created with all required SSM document for golden AMI automation and decommission.
9. CloudWatch event rules will be created that manages schedule run of inspector, execute SSM automation document and SSM AMI de-register document.
10. After completion of the Inspector Assessment run, SNS will publish to an email provided by user
11. Email will be sent out to the specified user with link to Inspector findings - assessment run name and links to either approve or deny the build after verifying findings
12. If user denied the build, golden AMI will be de-registered, and all created components will be deleted
13. If user approved the build, then golden AMI will be added to service catalog for organizational use
14. User can state if golden AMI needs accessed by cross accounts. User can provide list of Account ID's that require access

Conclusion

Golden AMI pipeline provides you with a solution for building, distributing and managing golden AMIs. If you are looking for a solution in AWS to distribute golden AMIs to various business units that continuously assesses the security posture and decommissions obsolete AMIs, our team of certified solutions architects can help. In today's fast paced world of technology, time to market is critical to business success. Enabling your teams to innovate at a faster pace will increase your chances of capturing more market share and outpacing your competition.