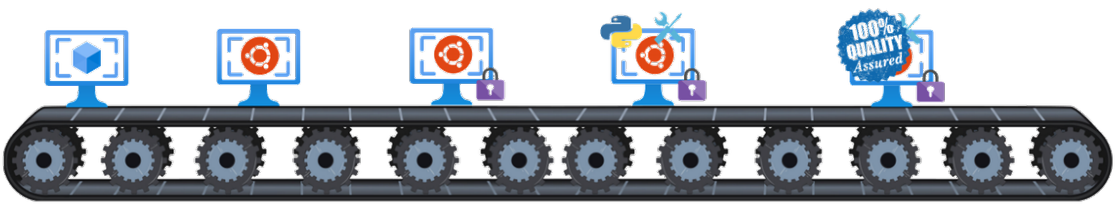
CONTINO

AUTOMATION

Azure Image Factory

Build Custom OS Image over Assured CIS Images



Azure Image Factory: Customised Images

- using GittHub repositories, GitHub actions, Terraform, Packer, Azure and Ansible to auto-build OS from pre-defined secured CIS Images either on demand or on os update

Requirement

Development Team
Business Project
Transformations

The organisation contains multiple elements with varying - but not considerably different requirements

Policies

our SecOps function have already created defintions and auto-updating OS versions for us to consume and build on

OS Choice

our Organisation has provided us with a pre-defined set of images that are up to standard, updated, patched and have the business tools and policies applied - we select any of the images available from the SecOps Shared Image Gallery as our source

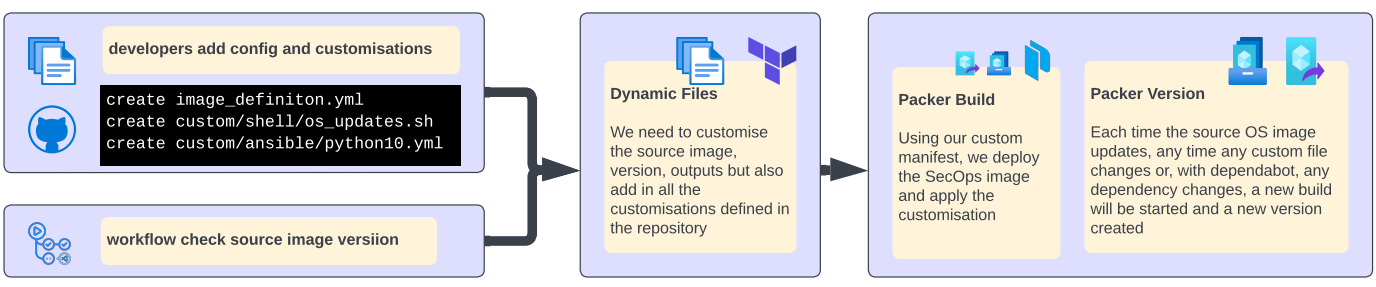
the development team create their definition - located in its own repository - that sets the required inputs to build a custom image

```
custom_dev = {
  input_gallery   = "Secops Image Gallery"
  input_image     = "0001-com-ubuntu-minimal-focal-minimal-20_04-lts-gen2"
  output_gallery  = "developer image gallery"
  output_image    = "focal_custom_python310"
  input_version   = "latest"
  output_version  = <semver>
}
```

ensuring the definition process is an "end user experience" without having to edit multiple files etc, the automation build process builds custom packer files to create the build - behind the scenes, we automatically find any files the developer has stored in ./custom, and we create the custom build

custom

When building, the packer process will run `./custom/shell/*` on the build VM - it will then install all `./custom/roles/*` and apply any `./custom/playbooks/*`
The key to this automation is that the developer simply supply verified custom steps



Azure Image Factory: Customised Images

- configurable by Azure Policy, Organistion OS Selection can be limited to secured base OS images, or images derived from them
- Custom Image definitions are delivered by self-service automation ensuring that the sources and workflows are configured to use internal galleries and configured to check for updates to their OS to trigger new builds
- Custom Image definitions are always built from the latest available updated OS (overridable)
- Custom Image Versions will have tooling and updates at least as often as the developer updates
- Custom Image Versions deliver new images, but do not - in the present configuration - deprecate old images
- Custom Image Versions are fully auditable - with metrics enabled, the evolution of images can be tracked

Repository (containing custom images) maintenence and image updates are managed by the requester and their team - with integrated communication and collaboration, custom images can become community managed