

# Checklist

## Compliance

### Repositories & Workflow Overview

- create source control elements
  - create 01-01-Compliance Base Infrastructure repository
    - workflow to run terraform on merge into main
  - create 01-02-Build Infrastructure repository
    - workflow to run when new build requested
    - build infrastructure persists for fixed period
    - scheduled workflow checks whether to destroy
  - create 01-03-Definitions repository
    - PR validates sources
    - PR test build
    - workflow run on updates to main

### Base Infrastructure

- Subscription
  - Service Principal
  - Resource Group
    - Key Vault
    - Storage Account
    - Image Gallery

### Image Gallery

```
terraform {
  backend "azurerm" {
    subscription_id = "625b66d7-5b11-40fb-99ab-ba303c13ea88"
    resource_group_name = "tf_state"
    storage_account_name = "continobakerytfstate"
    container_name = "example"
    key = "secops-infrastructure.tfstate"
  }
}
```

```
locals {
  ## details for creating the new additional shared image gallery in a new resource
  group (f enabled)
  location = "uksouth"
  shared_image_gallery_name = "secops_base_images"
  shared_image_gallery_resource_group_name = "secops_base_images-rg"
  shared_image_gallery_resource_group_create = true
  contributor_role_create = false
  reader_role_create = false
  managed_image_resource_group_name = "secops_base_images-rg"
}
```

```
data "azurerm_client_config" "current" {}

resource "azurerm_resource_group" "shared_image_gallery" {
  count = var.shared_image_gallery_resource_group_create == true ? 1 : 0

  name      = var.shared_image_gallery_resource_group_name
  location = var.location
}

data "azurerm_resource_group" "shared_image_gallery" {
  name = var.shared_image_gallery_resource_group_name

  depends_on = [ azurerm_resource_group.shared_image_gallery]
```

```

}

resource "azurerm_shared_image_gallery" "shared_image_gallery" {
  name                = var.shared_image_gallery_name
  resource_group_name = data.azurerm_resource_group.shared_image_gallery.name
  location            = data.azurerm_resource_group.shared_image_gallery.location
  description         = "Shared Images Gallery to store VMSS deployment images"

  lifecycle { ignore_changes = [ tags, ] }
}

```

## Build Infrastructure

### Build Resource Group

vNet

Subnet

NSG

Storage Account

## Packer Build Persisten Resources

```

## Build Image Factory Setup
## this variable file has been populated with all possible variables that could be
required by any configuration
## ity is intended to take use input variabilities and use logic with locals to determine
either a user set name or
## programatically created variable names - depending on the direction we take
variable "location" { default = "uksouth" }
variable "packer_build_resource_group_name" {}
variable "packer_build_resource_group_create" { default = true }
variable "packer_build_vnet_name" { }
variable "packer_build_vnet_address_space" { default = [ "10.0.0.0/16", ] }
variable "packer_build_vnet_create" { default = true }
variable "packer_build_subnet_name" { default = "packer_build" }
variable "packer_build_subnet_addresses" { default = [ "10.0.1.0/24", ] }

```

```
variable "packer_build_subnet_create" { default = true }
```

```
resource "azurerm_resource_group" "packer_build" {  
  #ts:skip=accurics.azure.NS.272 NoLock is required as resources will be temporary  
  count = var.packer_build_resource_group_create == true ? 1 : 0  
  
  name      = var.packer_build_resource_group_name  
  location = var.location  
}
```

```
resource "azurerm_virtual_network" "packer_build" {  
  count = var.packer_build_vnet_create == true ? 1 : 0  
  
  name                = var.packer_build_vnet_name  
  address_space       = var.packer_build_vnet_address_space  
  location             = var.location  
  resource_group_name = var.packer_build_resource_group_name  
  
  depends_on = [ azurerm_resource_group.packer_build,  
data.azurem_resource_group.packer_build ]  
}
```

```
data "azurerm_resource_group" "packer_build" {  
  name      = var.packer_build_resource_group_name  
  depends_on = [ azurerm_resource_group.packer_build ]  
}
```

```
data "azurerm_virtual_network" "packer_build" {  
  name                = var.packer_build_vnet_name  
  resource_group_name = var.packer_build_resource_group_name  
  depends_on         = [ azurerm_virtual_network.packer_build ]  
}
```

```
locals {  
  subnets = data.azurem_virtual_network.packer_build.subnets
```

```

}

resource "azurerm_subnet" "packer_build" {
  count = var.packer_build_subnet_create == true ? 1 : 0

  name                = var.packer_build_subnet_name
  resource_group_name = data.azurerm_virtual_network.packer_build.resource_group_name
  virtual_network_name = data.azurerm_virtual_network.packer_build.name
  address_prefixes    = var.packer_build_subnet_addresses

  depends_on = [ azurerm_virtual_network.packer_build ]
}

resource "azurerm_network_security_group" "packer_build" {
  count = var.packer_build_subnet_create == true ? 1 : 0

  ☐# checkov:skip=BC_AZR_NETWORKING_57: ADD REASON
  ☐# checkov:skip=BC_AZR_NETWORKING_2: ADD REASON
  ☐# checkov:skip=BC_AZR_NETWORKING_3: ADD REASON
  name                = "packer_nsg"
  location            = data.azurerm_resource_group.packer_build.location
  resource_group_name = data.azurerm_resource_group.packer_build.name

  security_rule {
    name                = "inallow"
    priority            = 100
    direction          = "Inbound"
    access              = "Allow"
    protocol            = "Tcp"
    source_port_range   = "*"
    destination_port_range = "*"
    source_address_prefix = "*"
    destination_address_prefix = "*"
  }
}

```

```

resource "azurerm_subnet_network_security_group_association" "packer_build" {
  count = var.packer_build_subnet_create == true ? 1 : 0

  subnet_id          = azurerm_subnet.packer_build.0.id
  network_security_group_id = azurerm_network_security_group.packer_build.0.id
}

resource "azurerm_network_security_rule" "packer_build_inbound" {
  count = var.packer_build_subnet_create == true ? 1 : 0

  name                = "sshin"
  priority            = 110
  direction           = "Inbound"
  access              = "Allow"
  protocol            = "Tcp"
  source_port_range   = "*"
  destination_port_range = "22"
  source_address_prefix = "*"
  destination_address_prefix = "*"
  resource_group_name =
azurerm_network_security_group.packer_build.0.resource_group_name
  network_security_group_name = azurerm_network_security_group.packer_build.0.name
}

resource "azurerm_network_security_rule" "packer_build_out" {
  name                = "allout"
  priority            = 110
  direction           = "Outbound"
  access              = "Allow"
  protocol            = "Tcp"
  source_port_range   = "*"
  destination_port_range = "*"
  source_address_prefix = "*"
  destination_address_prefix = "*"
  resource_group_name =

```

```
azurerm_network_security_group.packer_build.0.resource_group_name
  network_security_group_name = azurerm_network_security_group.packer_build.0.name
}
```

---

Revision #1

Created 10 January 2025 06:39:44 by pknw1

Updated 10 January 2025 07:30:42 by pknw1