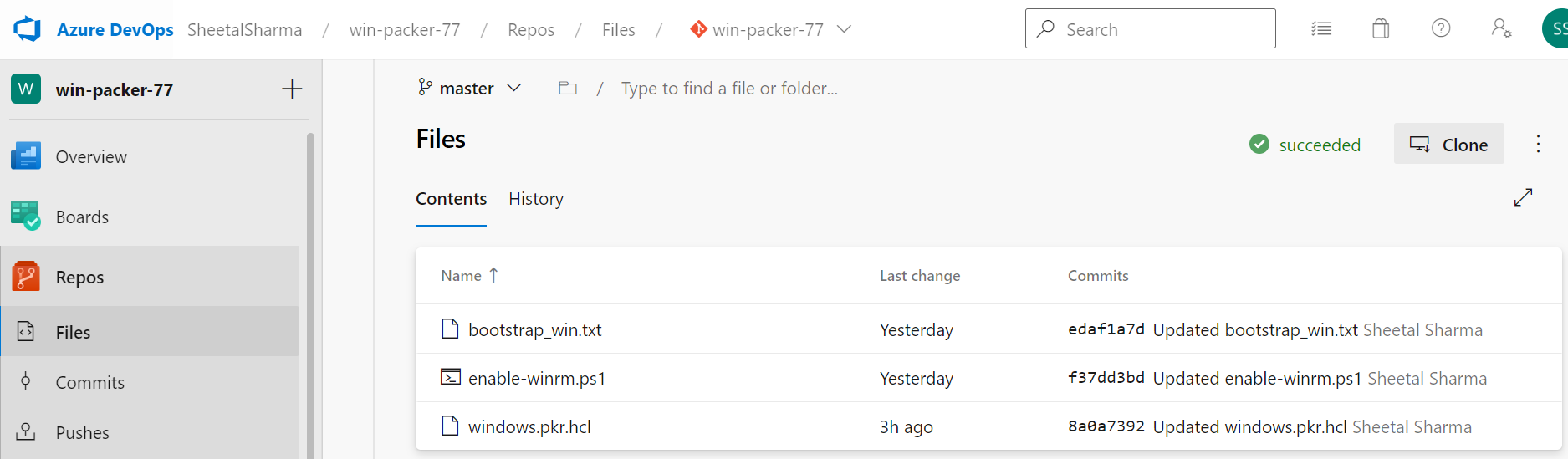
## **Build Windows AMI using Packer**

* Create Agent
* Go to project settings- Agent pool-New agent
* Create a repository in Azure



* Add the packer templates in that repo

## windows.pkr.hcl

packer {

  required\_plugins {

    amazon = {

      version = ">= 1.1.0"

      source = "github.com/hashicorp/amazon"

    }

  }

}

variable "region" {

  type    = string

  default = "us-east-1"

}

locals { timestamp = regex\_replace(timestamp(), "[- TZ:]", "") }

# source blocks are generated from your builders; a source can be referenced in

# build blocks. A build block runs provisioner and post-processors on a

# source.

source "amazon-ebs" "windows" {

  ami\_name      = "packer-windows-demo-${local.timestamp}"

  communicator  = "winrm"

  instance\_type = "t2.micro"

  region        = "${var.region}"

  source\_ami\_filter {

    filters = {

      name                = "Windows\_Server-2012-R2\*English-64Bit-Base\*"

      root-device-type    = "ebs"

      virtualization-type = "hvm"

    }

    most\_recent = true

    owners      = ["amazon"]

  }

  user\_data\_file = "./bootstrap\_win.txt"

  winrm\_password = "SuperS3cr3t!!!!"

  winrm\_username = "Administrator"

}

# a build block invokes sources and runs provisioning steps on them.

build {

  sources = ["source.amazon-ebs.windows"]

    provisioner "powershell" {

      environment\_vars = ["DEVOPS\_LIFE\_IMPROVER=PACKER"]

      inline           = ["Write-Host \"HELLO NEW USER; WELCOME TO $Env:DEVOPS\_LIFE\_IMPROVER\"", "Write-Host \"You need to use backtick escapes when using\"", "Write-Host \"characters such as DOLLAR`$ directly in a command\"", "Write-Host \"or in your own scripts.\""]

    }

    provisioner "powershell" {

      environment\_vars = ["VAR1=A$Dollar", "VAR2=A`Backtick", "VAR3=A'SingleQuote", "VAR4=A\"DoubleQuote"]

      script           = "./enable-winrm.ps1"

    }

    hcp\_packer\_registry {

    bucket\_name    = "packer-win3"

    bucket\_labels = {

      "owner"          = "platform-team"

      "os"             = "Windows",

      "windows-version" = "2022.08.10",

    }

    build\_labels = {

      "build-time"   = timestamp()

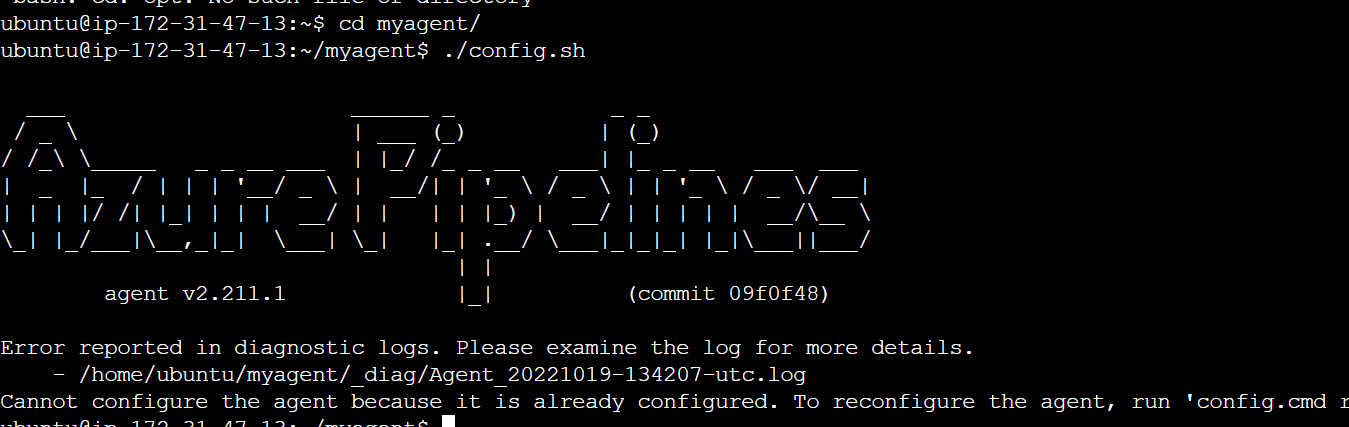
      "build-source" = basename(path.cwd)

    }

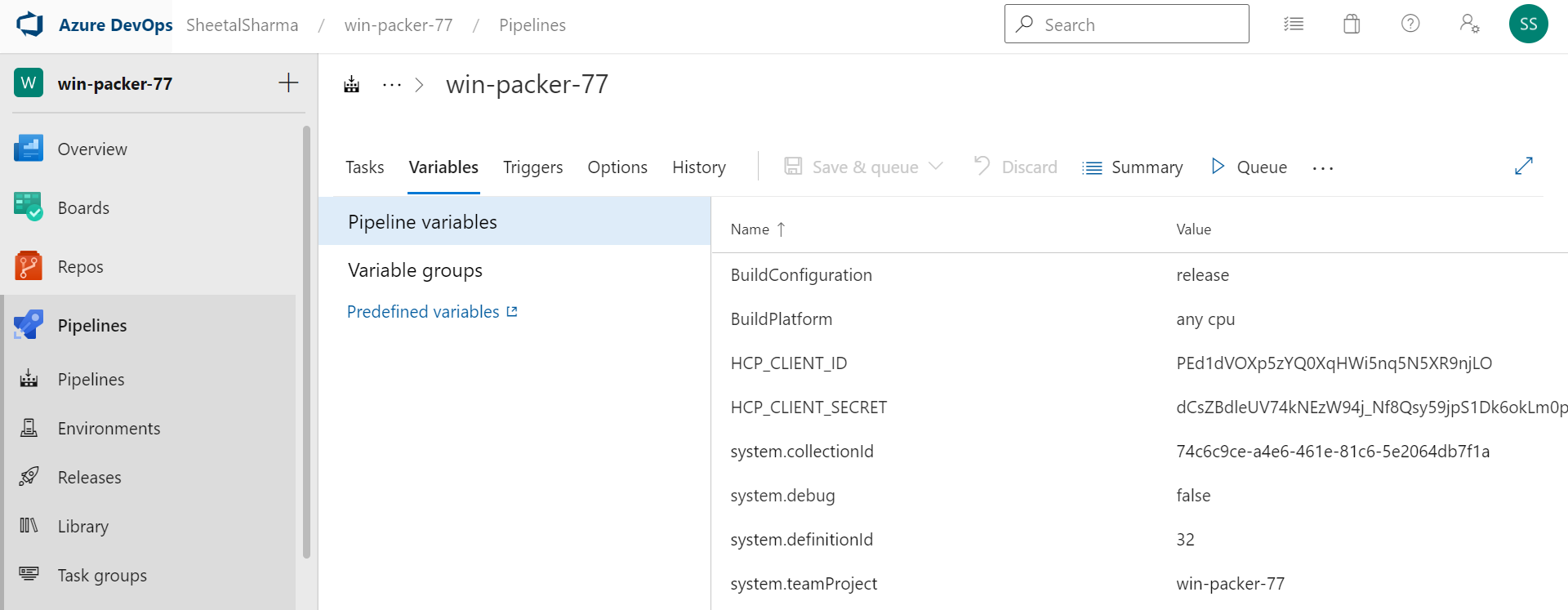
  }

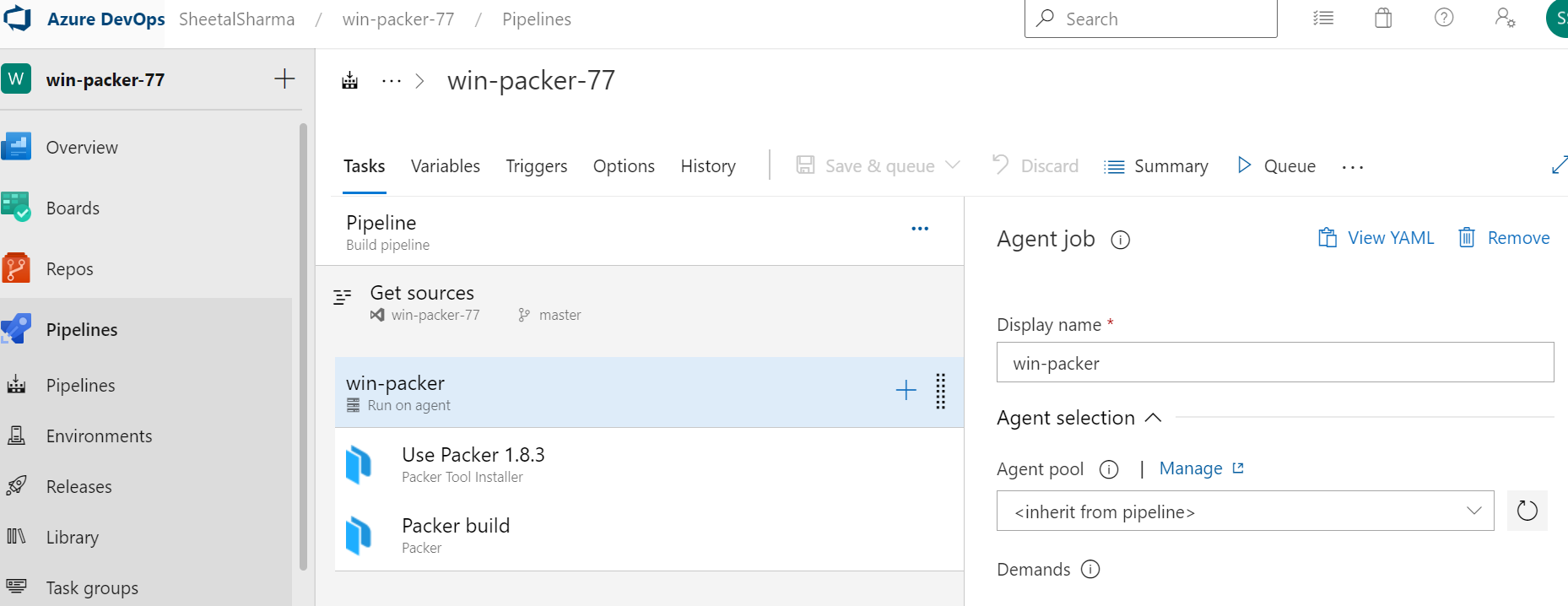
}

* Wget <https://vstsagentpackage.azureedge.net/agent/2.211.1/vsts-agent-linux-x64-2.211.1.tar.gz>
* mkdir myagent && cd myagent
* tar zxvf ~/Downloads/vsts-agent-linux-x64-2.211.1.tar.gz
* ./config.sh

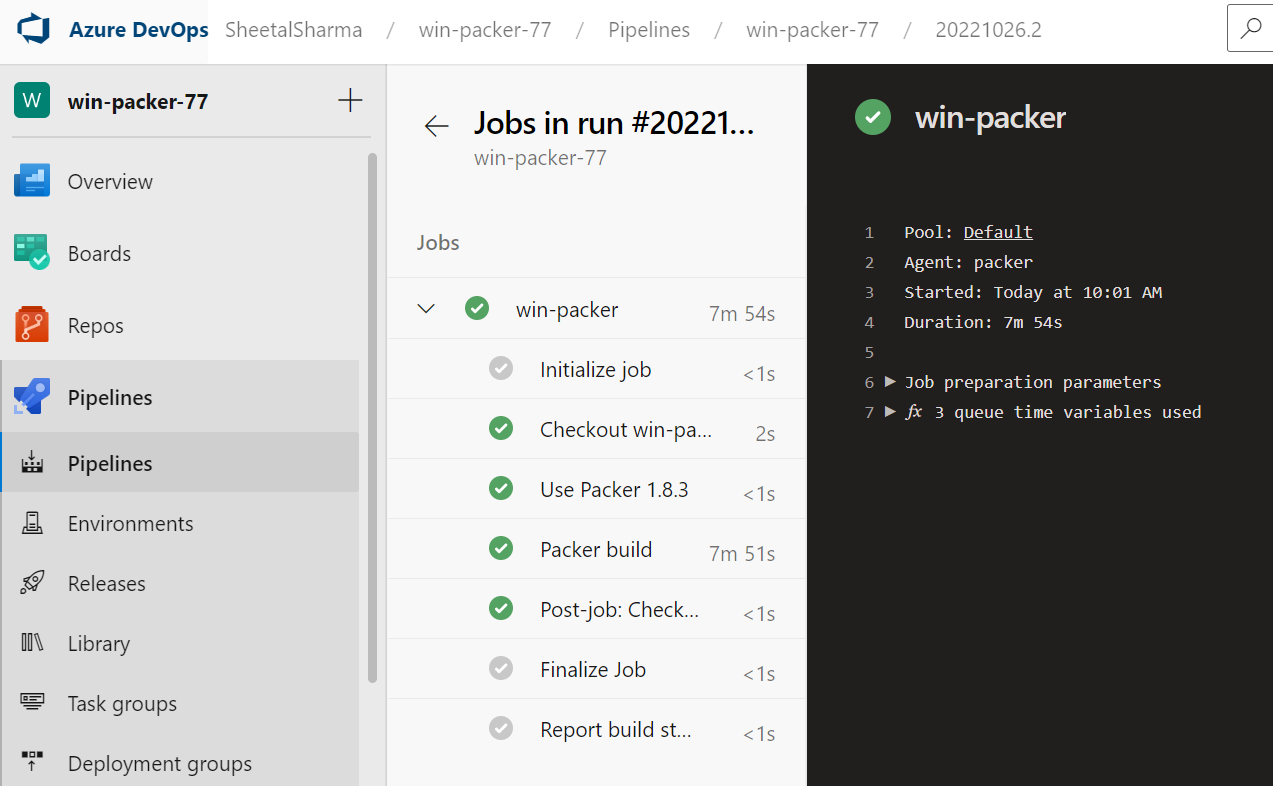


* ./run.sh
* Install AWS CLi in the instance
* Configure the Acess key and secret key
* To set environment variables Goto pipeline-library-variable group
* Add the variables





* Run Pipeline



* After running the pipeline AMI is created in the AWS account which we have configured and iteration stored in HCP registry.

