

Terraform init output

```
root@ip-172-31-51-14:/home/ubuntu/tf# terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/local...
- Finding latest version of hashicorp/tls...
- Installing hashicorp/local v2.2.3...
- Installed hashicorp/local v2.2.3 (signed by HashiCorp)
- Installing hashicorp/tls v4.0.4...
- Installed hashicorp/tls v4.0.4 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
root@ip-172-31-51-14:/home/ubuntu/tf#
```

Terraform plan output

```
root@ip-172-31-51-14:/home/ubuntu/tf# terraform plan
tls_private_key.priv-key: Refreshing state... [id=80d387316a3f26461f7a3c4266361c2a2b6df719]
local_file.id_rsa_games247: Refreshing state... [id=163800d279e0fc91165692d8df3777286d371420]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
<= read (data resources)

Terraform will perform the following actions:

# data.local_file.encoded-key will be read during apply
# (depends on a resource or a module with changes pending)
<= data "local_file" "encoded-key" {
  + content      = (known after apply)
  + content_base64 = (known after apply)
  + filename     = "/home/ubuntu/.ssh/id_rsa_games247"
  + id          = (known after apply)
}

# local_file.id_rsa_games247 will be created
+ resource "local_file" "id_rsa_games247" {
  + content      = (sensitive value)
  + directory_permission = "0777"
  + file_permission  = "0777"
  + filename       = "/home/ubuntu/.ssh/id_rsa_games247"
  + id           = (known after apply)
}

Plan: 1 to add, 0 to change, 0 to destroy.
```

```
Changes to Outputs:
~ base64-encoded-file = "L50tLS1CRUdJTB1SBU0EgUJFVJkFURSBLRVktLS0tL0pNSULKS0FQJkFBSNB20VBmHnyd3Mzdm1sa2VrOS8xeStNVGJaZ3BpWnndx0Q03VvphbWxkHBUXZNdKv5eG16Ck42WhhQ13pVv
09hUuNS2xo0FgaU1saZE2P31YU0JTMtPaTRk0r1RnZUdmVmskRPTXg223VjM0tceRaEXCWUkR11aeUvz2Ed0MOWeZHBVYVHM3MmR1Mk8rQnc4NkxJWRND2D1kxNVhB0ZFPOU9ncmRSbVb1Vadyexq21QQRVY0cApLbDhTR
z1NMWu3Q2RqenFPNkt6cEp1SUMrxdXVQcQtYmYmJ3SdqbWNRzSH1Hb1dIQW5KamM3QTFYUHNDRj1DeHNKckVXR0duTUN2dZg1dG54NUgVj3VJN3czUE1aUUYzaWf1R3VwZFdndKdVW1FTd0dVWG1QV1kVbWk42WpQMk1EbUsKM
Et1Wk3VVRURITG9ONjBRTE1yaWFEN1VGyWcNtNaUxe1E0SUNQazFy2kpQ0XJe1FKVFQ5YU0Sa9jNoT31PT3V0SgPRM0tqSkN6WnhRTGZ1YjFCRFmxS1g4Zk12a0IrNjRMNU0eUwzNU1WNV1sa1VvanBNMzVRM0JcT5Vh6cgnFod
jdfCmFnRk42dVF1c2ZEN0d6GpUMFJLTj1h2N1VQMOR1bWJ0cXVZU2ZNVn1YkzRIUVdBU01RaEZLTnpyVHLnCN9wK3oKUnN1TVQ3Wm1aWUzVWtseOFNYU1Q4aWZrT3U3dKlLc1JUTdYQXkZbHVcZmJRvMVeic1FwVWFSU0FuS
1hr2253YgppQ2UvdXhIbW0zdHJDY0tEbZjYTGf6eVrWzYtYdFBB21Y4S2pmR1ZDL0xe6aG5DXEzTD11cWqamhONW1NjNSC1NFQ1p6engvM1JRREF1TXZHTFFQd0dpYzBIN0orUXBjY0xt0WV2Smp3cEhVzTFT3zFbbjFDR
kpnLzkwQ0f3RUEKQVFLQfRnUFeeU0zVNCraVYK1a2xxdEE4121S2zPbTQ4YjQ0Y2RkZWVgc09w5j23dE1BRzYsKhq221N0QpXbXQ0THfncTruZW12NTY1R3dXWj1HWE8yZ294UDVcVwzWnNFUEy0p0pES9CR1FMST
W0b3bVWfdeCm42eFmCj1YHbCL0NZD23b9yZ2RmUPLU1d321dQMBsbqMhVv0a1NKRWVYnYranBDMFbMhMm45VDUKS0FQVDF1bVBzSUXLakprqTBMxkx1P25JWZdGepDuoDoeWTR1RkFfYcVXVjQ
Kd1cS91ace0NWXV3K3NMhJ05Qp44U5U1uRQa1VtKZuTmY3SfFNMWZ1Q2eEpdXpKaD1NRnF3RNMWdmsP2zBDMhMhOJQUT11WfVa3Y1T2RXYM1J1s1rUeP1VY1aeH9QmndwXJ0QzBKS11NzR1XTPXQjFrEX1S0WJd
W10Rn1ZaWwV6eJwTfKwCFOX1keP8KcKf0dXNBVZmZ11aXWJ3dV1Q31q1rStcVqNU4d2HYZ1MFS0eQa3BQCFEdmNN131Y21UHFJYnBvY01ac1tpdgp1cWfY4VYcYVhMhQWZ0tFR4U110c09Rm01a2xw0QpQVdH0TGRNb
C9rNDUDUXJ3Qk1jTGJUTGSMTX0a03p1TDbDCK1kWC9eGMyd2dmaGf3QngvYcW1oXVobG51dZyYcYjVVOThhUTRSem02Q11Nm11TFHhNk1La0ZQeVRND0d0UcKa1VrZVZBWSW1Ry9EQ1N1QVZ2cmP1e1d8C2H4U4weK84R
TR1MUyUPL184RHTJSTWtBkzcvK3F4XkdIUF2bVjVtUpkiWVh1NnFvbHFWbW9Xm1c4QVFRRTFYd2dZa1V50F1p2XBjWt1aU21Nq2hFU12DaloV5UVD22dFQkFQYmhnUD15ClhKUEPaY0E4TlFmVnMdh11NNWd0epVvZB3aEt4R
Ed221ZkWSstST1lyYbVtVbEpRSKqVdVoc1N0Wm1HSG0zWC8KS2pyeKvWDFXbHJDeJ2MaU1FV0pPK0qVjcxVjY6JMN5RVkNpCh11WkYyVWUyQndTTkpPT1BQTU2dXbKaFdubw0tmJCRKvPzU21ZmRVQXZFNkRDTjVpNGF5b
TNDT12kVDMk0j1nNm410E14dEP0UGmVHVUenpQSnFcm1JwZkY5C1lyRHVOsK1N1SHBvAjFHT05rM12hOHN1bVZNeVv1Zmt3U3g3M090Nm0z0T2ZempQaExXZ3R1QmHCK05LSDVRd1MKaJ1lSnK0U0o2eXhLd1RKv14WnUv
0hW0w1QfUB2U1JmZnczYm0wKkZW12c5eld4NvZWSHrVtWELMVmzrOWavNvphSGmxKzJmR21ZnhTK1VD22dFQkF0cm5KSzBcAVhV1Jot3A3Y2JzaUJWd25bznZQbmt5Ykk05k5YtTZHV1B7C1ZYT1YzQ122jNS01Yyc2tKW
XhpD1MOUEZJNGdRMMqcdHR2U5mYWNPSkpEV2pDT1F0RnVDMUdQdUpSaHFEbysKWFUySGJVFfPZennQg3F1Uj1wRFYyMTRz0UhcDBCWfHj1L3Uz2MvHcNNWZTbnNKR20o5aV10aDvR1BPRGFHQ2XmSwpYbGx2Und20xzdmdme
FRaR0dNWNhZb1Q53FMmnc1N0k5a0d6RytC05KRH1KMW0UbEJvc1haN12ObU9JQURyCktJQzJLLO2mL0t5WmF3aC9Xmd6V3U3M110T1YV3Evd1IweStUdmNmhR5jNre1h4bGZ1TTNNMFPremFVSHEKkHbTRdGdIL
zZFMGSETFhNRUpkdmf1U09420c1YX1Y21jYtnFQOTFka0NnZ0VBZG1KeUY4b2hncFpj1bW1BMQpRQXFCQnpFVTdURVQ451A3VjN4d1pW1YVjem220S40UUVkbj22dHvYcE1Ytnr0UhmBVexZjhFWj1CdGNRYjNMC1JxSHxkQ
2JgU2xuz23p1Nb1Yp91K1FYdnV5WpV13Y5dGk0RERpD4vUE9UtnV9Y2Y4bEfQZTfocvGHnj3BRXKYSKXZ4RDBWM252V1FvU9hQT13UjVhOUEya21T21JqY1RndjYzSUayVMAwejdXVX1td01ke1R1KMNVDcEFYM3h2Spwoc
E1lam9Td2122D0Tf1DNE9JbH2a2mXbKx6pT0laEHS1dTVHoy1dLdKZF0eT20QZuR04zST1CB1BnMkG1CNyXkUd1d1F8UKdMq3pW2WwWUZhSHeyQ0c3QnFwc11StYkpxURhRHFUR1EYVfgz0E5UaHJwMsvbvtZSjRWN
1YKw0eYfRSB0NBU0Y0tMk1p6CkXmEjK5QzE42mPbSNWzUxpdV1HU32YQYzUaRRa0ZKd3a1Uyd1Z1YpG51FCUVD2NTRjNGVMMW54mZyZkoved4UW83Y0R2KX1CbH9F3UjY45a1J3S3yETHY3bms5u2RjRW
kpx0LzC0cHb1J0WmnpqTHR1WHgYj1YhVgSStCfzBpY1A1CHY11dRcdraZHyac1pXZRS5BhmZN6G5Wn9QpFBCU0URK31G1J1R1k1WekKwHnTanVWkz5cXKqCtVYgmx5Gk3MQ3Y11Vc03sajd1eKRL1BNVUPa
G6NM1R5Q11Wmpe1BVR1R8dG9NvKVa0L2p1pOHdmdOVxNT121g1eSFBcK0UyUUNdmhvk70d11BSk4kMDUXS01nr2U4N1pVcKxYk13MEV2YXYxbCctra3B1TrJ1anY4bEcZdzFy2hCwQ9JQKFF0T1YzTj1SWEDeJlve
DHPa3dUNFNND2cNjgKwGF0HcwWJ0D2hmcVNVJFcxZExQZ1daWwJtZ10SEYyZT1lbkhdUTE20R3ExtVXESUth1S3RrOTB1dJNVTVObwpjcnhRd12REVMbMhXN1RWNV1V3cweHBEQjQcY3Z3SHVL1F80W1ZUmdD0d4W
E1ka0RUGf0a0YkWSvL3jP2CmFUTZFG6Yw1NR1FmNENVRJLzReD1JcFfNaN013aFvqXVGRGdV1VpT1h1UUVFTH1W0VUW2Y2yceEPL20KmtaWmpHqWgybzdx20d2Q3d6SGRDUN5GRGFTVSyehYkZkdNFnDNkN1U
F2mb2Yz3JUV0ZVU1F4Rnk1N1ENWpPd1ArVkfYVYVUalltL2arYzhwNGY4SUVMR1dq2YpxUGucW1xd1NP2m5Dd3p0YUgFS3B0WEzrUWV1R7OKL50tLS1FTKqgU1NBIFBSSV2BVEUG50VZLS0tLS0K" -> (known after
apply)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
root@ip-172-31-51-14:/home/ubuntu/tf#
```

Terraform apply output

```
root@ip-172-31-51-14:/home/ubuntu/tf# terraform apply --auto-approve
tls_private_key.priv-key: Refreshing state... [id=80d387316a3f26461f7a3c4266361c2a2b6df719]
local_file.id_rsa_games247: Refreshing state... [id=163800d279e0fc91165692d8df377286d371420]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
~ create
+ read (data resources)

Terraform will perform the following actions:

# data.local_file.encoded-key will be read during apply
# (depends on a resource or a module with changes pending)
<= data "local_file" "encoded-key" {
  + content           = (known after apply)
  + content_base64    = (known after apply)
  + filename          = "/home/ubuntu/.ssh/id_rsa_games247"
  + id                = (known after apply)
}

# local_file.id_rsa_games247 will be created
+ resource "local_file" "id_rsa_games247" {
  + content           = (sensitive value)
  + directory_permission = "0777"
  + file_permission    = "0777"
  + filename          = "/home/ubuntu/.ssh/id_rsa_games247"
  + id                = (known after apply)
}

Plan: 1 to add, 0 to change, 0 to destroy.
```

```
Changes to Outputs:
~ base64-encoded-file = "L50tLS1CRUdJTB1SBU0EgUJFVJkFURSBLRVktLS0tL0pNSULKS0FQJkFBSNB20VBmHnyd3Mzdm1sa2VrOS8xeStNVGJaZ3BpWnndx0Q03VvphbWxkHBUXZNdKv5eG16Ck42WhhQ13pVv
09hUuNS2xo0FgaU1saZE2P31YU0JTMtPaTRk0r1RnZUdmVmskRPTXg223VjM0tceRaEXCWUkR11aeUvz2Ed0MOWeZHBVYVHM3MmR1Mk8rQnc4NkxJWRND2D1kxNVhB0ZFPOU9ncmRSbVb1Vadyexq21QQRVY0cApLbDhTR
z1NMWu3Q2RqenFPNkt6cEp1SUMrxdXVQcQtYmYmJ3SdqbWNRzSH1Hb1dIQW5KamM3QTFYUHNDRj1DeHNKckVXR0duTUN2dZg1dG54NUgVj3VJN3czUE1aUUYzaWf1R3VwZFdndKdVW1FTd0dVWG1QV1kVbWk42WpQMk1EbUsKM
Et1Wk3VVRURITG9ONjBRTE1yaWFEN1VGyWcNtNaUxe1E0SUNQazFy2kpQ0XJe1FKVFQ5YU0Sa9jNoT31PT3V0SgPRM0tqSkN6WnhRTGZ1YjFCRFmxS1g4Zk12a0IrNjRMNU0eUwzNU1WNV1sa1VvanBNMzVRM0JcT5Vh6cgnFod
jdfCmFnRk42dVF1c2ZEN0d6GpUMFJLTj1h2N1VQMOR1bWJ0cXVZU2ZNVn1YkzRIUVdBU01RaEZLTnpyVHLnCN9wK3oKUnN1TVQ3Wm1aWUzVWtseOFNYU1Q4aWZrT3U3dKlLc1JUTdYQXkZbHVcZmJRvMVeic1FwVWFSU0FuS
1hr2253YgppQ2UvdXhIbW0zdHJDY0tEbZjYTGf6eVrWzYtYdFBB21Y4S2pmR1ZDL0xe6aG5DXEzTD11cWqamhONW1NjNSC1NFQ1p6engvM1JRREF1TXZHTFFQd0dpYzBIN0orUXBjY0xt0WV2Smp3cEhVzTFT3zFbbjFDR
kpnLzkwQ0f3RUEKQVFLQfRnUFeeU0zVNCraVYK1a2xxdEE4121S2zPbTQ4YjQ0Y2RkZWVgc09w5j23dE1BRzYsKhq221N0QpXbXQ0THfncTruZW12NTY1R3dXWj1HWE8yZ294UDVcVwzWnNFUEy0p0pES9CR1FMST
W0b3bVWfdeCm42eFmCj1YHbCL0NZD23b9yZ2RmUPLU1d321dQMBsbqMhVv0a1NKRWVYnYranBDMFbMhMm45VDUKS0FQVDF1bVBzSUXLakprqTBMxkx1P25JWZdGepDuoDoeWTR1RkFfYcVXVjQ
Kd1cS91ace0NWXV3K3NMhJ05Qp44U5U1uRQa1VtKZuTmY3SfFNMWZ1Q2eEpdXpKaD1NRnF3RNMWdmsP2zBDMhMhOJQUT11WfVa3Y1T2RXYM1J1s1rUeP1VY1aeH9QmndwXJ0QzBKS11NzR1XTPXQjFrEX1S0WJd
W10Rn1ZaWwV6eJwTfKwCFOX1keP8KcKf0dXNBVZmZ11aXWJ3dV1Q31q1rStcVqNU4d2HYZ1MFS0eQa3BQCFEdmNN131Y21UHFJYnBvY01ac1tpdgp1cWfY4VYcYVhMhQWZ0tFR4U110c09Rm01a2xw0QpQVdH0TGRNb
C9rNDUDUXJ3Qk1jTGJUTGSMTX0a03p1TDbDCK1kWC9eGMyd2dmaGf3QngvYcW1oXVobG51dZyYcYjVVOThhUTRSem02Q11Nm11TFHhNk1La0ZQeVRND0d0UcKa1VrZVZBWSW1Ry9EQ1N1QVZ2cmP1e1d8C2H4U4weK84R
TR1MUyUPL184RHTJSTWtBkzcvK3F4XkdIUF2bVjVtUpkiWVh1NnFvbHFWbW9Xm1c4QVFRRTFYd2dZa1V50F1p2XBjWt1aU21Nq2hFU12DaloV5UVD22dFQkFQYmhnUD15ClhKUEPaY0E4TlFmVnMdh11NNWd0epVvZB3aEt4R
Ed221ZkWSstST1lyYbVtVbEpRSKqVdVoc1N0Wm1HSG0zWC8KS2pyeKvWDFXbHJDeJ2MaU1FV0pPK0qVjcxVjY6JMN5RVkNpCh11WkYyVWUyQndTTkpPT1BQTU2dXbKaFdubw0tmJCRKvPzU21ZmRVQXZFNkRDTjVpNGF5b
TNDT12kVDMk0j1nNm410E14dEP0UGmVHVUenpQSnFcm1JwZkY5C1lyRHVOsK1N1SHBvAjFHT05rM12hOHN1bVZNeVv1Zmt3U3g3M090Nm0z0T2ZempQaExXZ3R1QmHCK05LSDVRd1MKaJ1lSnK0U0o2eXhLd1RKv14WnUv
0hW0w1QfUB2U1JmZnczYm0wKkZW12c5eld4NvZWSHrVtWELMVmzrOWavNvphSGmxKzJmR21ZnhTK1VD22dFQkF0cm5KSzBcAVhV1Jot3A3Y2JzaUJWd25bznZQbmt5Ykk05k5YtTZHV1B7C1ZYT1YzQ122jNS01Yyc2tKW
XhpD1MOUEZJNGdRMMqcdHR2U5mYWNPSkpEV2pDT1F0RnVDMUdQdUpSaHFEbysKWFUySGJVFfPZennQg3F1Uj1wRFYyMTRz0UhcDBCWfHj1L3Uz2MvHcNNWZTbnNKR20o5aV10aDvR1BPRGFHQ2XmSwpYbGx2Und20xzdmdme
FRaR0dNWNhZb1Q53FMmnc1N0k5a0d6RytC05KRH1KMW0UbEJvc1haN12ObU9JQURyCktJQzJLLO2mL0t5WmF3aC9Xmd6V3U3M110T1YV3Evd1IweStUdmNmhR5jNre1h4bGZ1TTNNMFPremFVSHEKkHbTRdGdIL
zZFMGSETFhNRUpkdmf1U09420c1YX1Y21jYtnFQOTFka0NnZ0VBZG1KeUY4b2hncFpj1bW1BMQpRQXFCQnpFVTdURVQ451A3VjN4d1pW1YVjem220S40UUVkbj22dHvYcE1Ytnr0UhmBVexZjhFWj1CdGNRYjNMC1JxSHxkQ
2JgU2xuz23p1Nb1Yp91K1FYdnV5WpV13Y5dGk0RERpD4vUE9UtnV9Y2Y4bEfQZTfocvGHnj3BRXKYSKXZ4RDBWM252V1FvU9hQT13UjVhOUEya21T21JqY1RndjYzSUayVMAwejdXVX1td01ke1R1KMNVDcEFYM3h2Spwoc
E1lam9Td2122D0Tf1DNE9JbH2a2mXbKx6pT0laEHS1dTVHoy1dLdKZF0eT20QZuR04zST1CB1BnMkG1CNyXkUd1d1F8UKdMq3pW2WwWUZhSHeyQ0c3QnFwc11StYkpxURhRHFUR1EYVfgz0E5UaHJwMsvbvtZSjRWN
1YKw0eYfRSB0NBU0Y0tMk1p6CkXmEjK5QzE42mPbSNWzUxpdV1HU32YQYzUaRRa0ZKd3a1Uyd1Z1YpG51FCUVD2NTRjNGVMMW54mZyZkoved4UW83Y0R2KX1CbH9F3UjY45a1J3S3yETHY3bms5u2RjRW
kpx0LzC0cHb1J0WmnpqTHR1WHgYj1YhVgSStCfzBpY1A1CHY11dRcdraZHyac1pXZRS5BhmZN6G5Wn9QpFBCU0URK31G1J1R1k1WekKwHnTanVWkz5cXKqCtVYgmx5Gk3MQ3Y11Vc03sajd1eKRL1BNVUPa
G6NM1R5Q11Wmpe1BVR1R8dG9NvKVa0L2p1pOHdmdOVxNT121g1eSFBcK0UyUUNdmhvk70d11BSk4kMDUXS01nr2U4N1pVcKxYk13MEV2YXYxbCctra3B1TrJ1anY4bEcZdzFy2hCwQ9JQKFF0T1YzTj1SWEDeJlve
DHPa3dUNFNND2cNjgKwGF0HcwWJ0D2hmcVNVJFcxZExQZ1daWwJtZ10SEYyZT1lbkhdUTE20R3ExtVXESUth1S3RrOTB1dJNVTVObwpjcnhRd12REVMbMhXN1RWNV1V3cweHBEQjQcY3Z3SHVL1F80W1ZUmdD0d4W
E1ka0RUGf0a0YkWSvL3jP2CmFUTZFG6Yw1NR1FmNENVRJLzReD1JcFfNaN013aFvqXVGRGdV1VpT1h1UUVFTH1W0VUW2Y2yceEPL20KmtaWmpHqWgybzdx20d2Q3d6SGRDUN5GRGFTVSyehYkZkdNFnDNkN1U
F2mb2Yz3JUV0ZVU1F4Rnk1N1ENWpPd1ArVkfYVYVUalltL2arYzhwNGY4SUVMR1dq2YpxUGucW1xd1NP2m5Dd3p0YUgFS3B0WEzrUWV1R7OKL50tLS1FTKqgU1NBIFBSSV2BVEUG50VZLS0tLS0K" -> (known after
apply)

local_file.id_rsa_games247: Creating...
local_file.id_rsa_games247: Creation complete after 0s [id=163800d279e0fc91165692d8df377286d371420]
data.local_file.encoded-key: Reading...
data.local_file.encoded-key: Read complete after 0s [id=163800d279e0fc91165692d8df377286d371420]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

```
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

Outputs:

[illegible]