```
# Deploying OneAgent

> **Series:** ONBRD | **Notebook:** 5 of 10 | **Created:** December 2025

## Getting Data Into Dynatrace

OneAgent is the foundation of Dynatrace monitoring. This notebook covers
deployment strategies, installation methods, and verification steps to ensure
your infrastructure is reporting data.

---

## Table of Contents

1. What is OneAgent?
2. Deployment Strategy
3. Generating a Deployment Token
4. Installation Methods
5. Kubernetes Deployment
6. Verifying Deployment
7. Troubleshooting
8. Next Steps

---

## Prerequisites

- Admin access to Dynatrace environment
- API token with `InstallerDownload` scope (created in ONBRD-02)
- Root/admin access to target hosts
- Network access from hosts to Dynatrace (port 443) or ActiveGate (ONBRD-03)
- Cloud integrations configured (ONBRD-04) for full context

## 1. What is OneAgent?

OneAgent is Dynatrace's unified monitoring agent that:

| Capability | Description |
|------------|-------------|
| **Auto-discovery** | Automatically detects hosts, processes, services |
| **Full-stack** | Infrastructure, application, and user experience |
| **Zero configuration** | Works immediately after installation |
| **Distributed tracing** | End-to-end transaction visibility |
| **Log collection** | Collects and forwards log data |

![OneAgent Architecture]
(data:image/svg+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdm
ciIHZpZXdCb3g9IjAgMCA4MDAgMzQwIj4KICA8ZGVmcz4KICAgIDxsaW5lYXJHcmFkaWVudCBpZD0
iaG9zdEdyYWQiIHgxPSIwJSIgeTE9IjAlIiB4Mj0iMTAwJSIgeTI9IjEwMCUiPgogICAgICA8c3Rv
cCBvZmZzZXQ9IjAlIiBzdHlsZT0ic3RvcC1jb2xvcjojNjQ3NDhiO3N0b3Atb3BhY2l0eToxIiAvP
gogICAgICA8c3RvcCBvZmZzZXQ9IjEwMCUiIHN0eWxlPSJzdG9wLWNvbG9yOiM0NzU1Njk7c3RvcC
1vcGFjaXR5OjEiIC8+CiAgICA8L2xpbmVhckdyYWRpZW50PgogICAgPGxpbmVhckdyYWRpZW50IGl
kPSJhZ2VudEdyYWQiIHgxPSIwJSIgeTE9IjAlIiB4Mj0iMTAwJSIgeTI9IjEwMCUiPgogICAgICA8
c3RvcCBvZmZzZXQ9IjAlIiBzdHlsZT0ic3RvcC1jb2xvcjojMTQ5NmZmO3N0b3Atb3BhY2l0eToxI
iAvPgogICAgICA8c3RvcCBvZmZzZXQ9IjEwMCUiIHN0eWxlPSJzdG9wLWNvbG9yOiMwYTY0YmM7c3
RvcC1vcGFjaXR5OjEiIC8+CiAgICA8L2xpbmVhckdyYWRpZW50PgogICAgPGxpbmVhckdyYWRpZW5
0IGlkPSJwcm9jZXNzR3JhZCIgeDE9IjAlIiB5MT0iMCUiIHgyPSIxMDAlIiB5Mj0iMTAwJSI+CiAg
ICAgIDxzdG9wIG9mZnNldD0iMCUiIHN0eWxlPSJzdG9wLWNvbG9yOiMxMGI5ODE7c3RvcC1vcGFja
XR5OjEiIC8+CiAgICAgIDxzdG9wIG9mZnNldD0iMTAwJSIgc3R5bGU9InN0b3AtY29sb3I6IzA1OT
Y2OTtzdG9wLW9wYWNpdHk6MSIgLz4KICAgIDwvbGluZWFyR3JhZGllbnQ+CiAgICA8bGluZWFyR3J
hZGllbnQgaWQ9ImNsdXN0ZXJHcmFkIiB4MT0iMCUiIHkxPSIwJSIgeDI9IjEwMCUiIHkyPSIxMDAl
Ij4KICAgICAgPHN0b3Agb2Zmc2V0PSIwJSIgc3R5bGU9InN0b3AtY29sb3I6IzhiNWNmNjtzdG9wL
W9wYWNpdHk6MSIgLz4KICAgICAgPHN0b3Agb2Zmc2V0PSIxMDAlIiBzdHlsZT0ic3RvcC1jb2xvcj
ojN2MzYWVkO3N0b3Atb3BhY2l0eToxIiAvPgogICAgPC9saW5lYXJHcmFkaWVudD4KICAgIDxmaWx
0ZXIgaWQ9ImFnZW50U2hhZG93Ij4KICAgICAgPGZlRHJvcFNoYWRvdyBkeD0iMSIgZHk9IjEiIHN0
ZERldmlhdGlvbj0iMiIgZmxvb2Qtb3BhY2l0eT0iMC4xNSIvPgogICAgPC9maWx0ZXI+CiAgICA8b
WFya2VyIGlkPSJhZ2VudEFycm93IiBtYXJrZXJXaWR0aD0iMTAiIG1hcmtlckhlaWdodD0iNyIgcm
VmWD0iOSIgcmVmWT0iMy41IiBvcmllbnQ9ImF1dG8iPgogICAgICA8cG9seWdvbiBwb2ludHM9IjA
gMCwgMTAgMy41LCAwIDciIGZpbGw9IiM2NDc0OGIiLz4KICAgIDwvbWFya2VyPgogIDwvZGVmcz4K
CiAgPCEtLSBCYWNrZ3JvdW5kIC0tPgogIDxyZWN0IHdpZHRoPSI4MDAiIGhlaWdodD0iMzQwIiBma
WxsPSIjZjhmOWZhIiByeD0iMTAiLz4KCiAgPCEtLSBUaXRsZSAtLT4KICA8dGV4dCB4PSI0MDAiIH
k9IjI4IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6ZT0iMTgiIGZvbnQ
td2VpZ2h0PSJib2xkIiBmaWxsPSIjMzMzIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5PbmVBZ2VudCBB
cmNoaXRlY3R1cmU8L3RleHQ+CiAgPHRleHQgeD0iNDAwIiB5PSI0OCIgZm9udC1mYW1pbHk9IkFya
WFsLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjExIiBmaWxsPSIjNjY2IiB0ZXh0LWFuY2hvcj0ibW
lkZGxlIj5VbmlmaWVkIGFnZW50IGZvciBmdWxsLXN0YWNrIG1vbml0b3Jpbmc8L3RleHQ+CgogIDw
hLS0gSG9zdCBDb250YWluZXIgLS0+CiAgPHJlY3QgeD0iNDAiIHk9IjcwIiB3aWR0aD0iNTIwIiBo
ZWlnaHQ9IjIyMCIgcng9IjEyIiBmaWxsPSIjZmZmIiBzdHJva2U9IiM2NDc0OGIiIHN0cm9rZS13a
WR0aD0iMiIgZmlsdGVyPSJ1cmwoI2FnZW50U2hhZG93KSIvPgogIDxyZWN0IHg9IjQwIiB5PSI3MC
Igd2lkdGg9IjUyMCIgaGVpZ2h0PSIzNSIgcng9IjEyIiBmaWxsPSJ1cmwoI2hvc3RHcmFkKSIvPgo
gIDxyZWN0IHg9IjQwIiB5PSI5NSIgd2lkdGg9IjUyMCIgaGVpZ2h0PSIxMCIgZmlsbD0idXJsKCNo
b3N0R3JhZCkiLz4KICA8dGV4dCB4PSIzMDAiIHk9Ijk1IiBmb250LWZhbWlseT0iQXJpYWwsIHNhb
nMtc2VyaWYiIGZvbnQtc2l6ZT0iMTQiIGZvbnQtd2VpZ2h0PSJib2xkIiBmaWxsPSJ3aGl0ZSIgdG
V4dC1hbmNob3I9Im1pZGRsZSI+SE9TVDwvdGV4dD4KCiAgPCEtLSBPbmVBZ2VudCBDb250YWluZXI
gLS0+CiAgPHJlY3QgeD0iNjAiIHk9IjExNSIgd2lkdGg9IjQ4MCIgaGVpZ2h0PSIxNjAiIHJ4PSIx
MCIgZmlsbD0iI2VmZjZmZiIgc3Ryb2tlPSIjMTQ5NmZmIiBzdHJva2Utd2lkdGg9IjIiLz4KICA8c
mVjdCB4PSI2MCIgeT0iMTE1IiB3aWR0aD0iNDgwIiBoZWlnaHQ9IjI4IiByeD0iMTAiIGZpbGw9In
VybCgjYWdlbnRHcmFkKSIvPgogIDxyZWN0IHg9IjYwIiB5PSIxMzUiIHdpZHRoPSI0ODAiIGhlaWd
odD0iOCIgZmlsbD0idXJsKCNhZ2VudEdyYWQpIi8+CiAgPHRleHQgeD0iMzAwIiB5PSIxMzUiIGZv
bnQtZmFtaWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMiIgZm9udC13ZWlnaHQ9I
mJvbGQiIGZpbGw9IndoaXRlIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5PbmVBZ2VudDwvdGV4dD4KCi
AgPCEtLSBQcm9jZXNzIE1vbml0b3JzIC0tPgogIDxyZWN0IHg9IjgwIiB5PSIxNTUiIHdpZHRoPSI
xNDAiIGhlaWdodD0iNTAiIHJ4PSI2IiBmaWxsPSJ1cmwoI3Byb2Nlc3NHcmFkKSIvPgogIDx0ZXh0
IHg9IjE1MCIgeT0iMTc3IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6Z
T0iMTAiIGZvbnQtd2VpZ2h0PSJib2xkIiBmaWxsPSJ3aGl0ZSIgdGV4dC1hbmNob3I9Im1pZGRsZS
```

```
I+UHJvY2VzczwvdGV4dD4KICA8dGV4dCB4PSIxNTAiIHk9IjE5MiIgZm9udC1mYW1pbHk9IkFyaWF
sLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjEwIiBmaWxsPSJyZ2JhKDI1NSwyNTUsMjU1LDAuOSki
IHRleHQtYW5jaG9yPSJtaWRkbGUiPk1vbml0b3Jpbmc8L3RleHQ+CgogIDxyZWN0IHg9IjIzNSIge
T0iMTU1IiB3aWR0aD0iMTQwIiBoZWlnaHQ9IjUwIiByeD0iNiIgZmlsbD0idXJsKCNwcm9jZXNzR3
JhZCkiLz4KICA8dGV4dCB4PSIzMDUiIHk9IjE3NyIgZm9udC1mYW1pbHk9IkFyaWFsLCBzYW5zLXN
lcmlmIiBmb250LXNpemU9IjEwIiBmb250LXdlaWdodD0iYm9sZCIgZmlsbD0id2hpdGUiIHRleHQt
YW5jaG9yPSJtaWRkbGUiPlByb2Nlc3M8L3RleHQ+CiAgPHRleHQgeD0iMzA1IiB5PSIxOTIiIGZvb
nQtZmFtaWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMCIgZmlsbD0icmdiYSgyNT
UsMjU1LDI1NSwwLjkpIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5Nb25pdG9yaW5nPC90ZXh0PgoKICA
8cmVjdCB4PSIzOTAiIHk9IjE1NSIgd2lkdGg9IjE0MCIgaGVpZ2h0PSI1MCIgcng9IjYiIGZpbGw9
InVybCgjcHJvY2Vzc0dyYWQpIi8+CiAgPHRleHQgeD0iNDYwIiB5PSIxNzciIGZvbnQtZmFtaWx5P
SJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMCIgZm9udC13ZWlnaHQ9ImJvbGQiIGZpbG
w9IndoaXRlIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5Qcm9jZXNzPC90ZXh0PgogIDx0ZXh0IHg9IjQ
2MCIgeT0iMTkyIiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6ZT0iMTAi
IGZpbGw9InJnYmEoMjU1LDI1NSwyNTUsMC45KSIgdGV4dC1hbmNob3I9Im1pZGRsZSI+TW9uaXRvc
mluZzwvdGV4dD4KCiAgPCEtLSBJbmZyYXN0cnVjdHVyZSBNb25pdG9yaW5nIC0tPgogIDxyZWN0IH
g9IjgwIiB5PSIyMTUiIHdpZHRoPSIzMDAiIGhlaWdodD0iMzUiIHJ4PSI2IiBmaWxsPSIjZGJlYWZ
lIiBzdHJva2U9IiMzYjgyZjYiIHN0cm9rZS13aWR0aD0iMSIvPgogIDx0ZXh0IHg9IjIzMCIgeT0i
MjM4IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6ZT0iMTEiIGZpbGw9I
iMxZTQwYWYiIHRleHQtYW5jaG9yPSJtaWRkbGUiPkluZnJhc3RydWN0dXJlIE1vbml0b3JpbmcgKE
NQVSwgTWVtb3J5LCBEaXNrKTwvdGV4dD4KCiAgPCEtLSBMb2cgQ29sbGVjdGlvbiAtLT4KICA8cmV
jdCB4PSIzOTAiIHk9IjIxNSIgd2lkdGg9IjE0MCIgaGVpZ2h0PSIzNSIgcng9IjYiIGZpbGw9IiNm
ZWYzYzciIHN0cm9rZT0iI2Y1OWUwYiIgc3Ryb2tlLXdpZHRoPSIxIi8+CiAgPHRleHQgeD0iNDYwI
iB5PSIyMzgiIGZvbnQtZmFtaWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMSIgZm
lsbD0iIzkyNDAwZSIgdGV4dC1hbmNob3I9Im1pZGRsZSI+TG9nIENvbGxlY3Rpb248L3RleHQ+Cgo
gIDwhLS0gQ29ubmVjdGlvbiBBcnJvdyAtLT4KICA8cGF0aCBkPSJNNTYwLDE4MCBMNjMwLDE4MCIg
c3Ryb2tlPSIjNjQ3NDhiIiBzdHJva2Utd2lkdGg9IjMiIGZpbGw9Im5vbmUiIG1hcmtlci1lbmQ9I
nVybCgjYWdlbnRBcnJvdykiLz4KICA8dGV4dCB4PSI1OTUiIHk9IjE3MCIgZm9udC1mYW1pbHk9Ik
FyaWFsLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjEwIiBmaWxsPSIjNjQ3NDhiIiB0ZXh0LWFuY2h
vcj0ibWlkZGxlIj5IVFRQUzwvdGV4dD4KICA8dGV4dCB4PSI1OTUiIHk9IjE5NSIgZm9udC1mYW1p
bHk9IkFyaWFsLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjEwIiBmaWxsPSIjNjQ3NDhiIiB0ZXh0L
WFuY2hvcj0ibWlkZGxlIj4oNDQzKTwvdGV4dD4KCiAgPCEtLSBEeW5hdHJhY2UgQ2x1c3RlciAtLT
4KICA8cmVjdCB4PSI2NDAiIHk9IjEyMCIgd2lkdGg9IjE0MCIgaGVpZ2h0PSIxMjAiIHJ4PSIxMCI
gZmlsbD0iI2ZmZiIgc3Ryb2tlPSIjOGI1Y2Y2IiBzdHJva2Utd2lkdGg9IjIiIGZpbHRlcj0idXJs
KCNhZ2VudFNoYWRvdykiLz4KICA8cmVjdCB4PSI2NDAiIHk9IjEyMCIgd2lkdGg9IjE0MCIgaGVpZ
2h0PSIzMCIgcng9IjEwIiBmaWxsPSJ1cmwoI2NsdXN0ZXJHcmFkKSIvPgogIDxyZWN0IHg9IjY0MC
IgeT0iMTQyIiB3aWR0aD0iMTQwIiBoZWlnaHQ9IjgiIGZpbGw9InVybCgjY2x1c3RlckdyYWQpIi8
+CiAgPHRleHQgeD0iNzEwIiB5PSIxNDIiIGZvbnQtZmFtaWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIg
Zm9udC1zaXplPSIxMSIgZm9udC13ZWlnaHQ9ImJvbGQiIGZpbGw9IndoaXRlIiB0ZXh0LWFuY2hvc
j0ibWlkZGxlIj5EeW5hdHJhY2U8L3RleHQ+CiAgPHRleHQgeD0iNzEwIiB5PSIxNzUiIGZvbnQtZm
FtaWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMCIgZmlsbD0iIzY0NzQ4YiIgdGV
4dC1hbmNob3I9Im1pZGRsZSI+Q2x1c3RlcjwvdGV4dD4KICA8dGV4dCB4PSI3MTAiIHk9IjE5NSIg
Zm9udC1mYW1pbHk9IkFyaWFsLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjEwIiBmaWxsPSIjNjQ3N
DhiIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5BbmFseXNpczwvdGV4dD4KICA8dGV4dCB4PSI3MTAiIH
k9IjIxNSIgZm9udC1mYW1pbHk9IkFyaWFsLCBzYW5zLXNlcmlmIiBmb250LXNpemU9IjEwIiBmaWx
sPSIjNjQ3NDhiIiB0ZXh0LWFuY2hvcj0ibWlkZGxlIj5TdG9yYWdlPC90ZXh0PgogIDx0ZXh0IHg9
IjcxMCIgeT0iMjM1IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6ZT0iM
TAiIGZpbGw9IiM2NDc0OGIiIHRleHQtYW5jaG9yPSJtaWRkbGUiPkFsZXJ0aW5nPC90ZXh0PgoKIC
```

```
A8IS0tIENhcGFiaWxpdGllcyBMZWdlbmQgLS0+CiAgPHJlY3QgeD0iNDAiIHk9IjMwMCIgd2lkdGg
9IjcyMCIgaGVpZ2h0PSIzMCIgcng9IjYiIGZpbGw9IiNmZmYiIHN0cm9rZT0iI2UyZThmMCIgc3Ry
b2tlLXdpZHRoPSIxIi8+CiAgPGNpcmNsZSBjeD0iNzAiIGN5PSIzMTUiIHI9IjgiIGZpbGw9IiMxM
GI5ODEiLz4KICA8dGV4dCB4PSI4NSIgeT0iMzE5IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2
VyaWYiIGZvbnQtc2l6ZT0iMTAiIGZpbGw9IiMzMzMiPkF1dG8tZGlzY292ZXJ5PC90ZXh0PgogIDx
jaXJjbGUgY3g9IjE4NSIgY3k9IjMxNSIgcj0iOCIgZmlsbD0iIzNiODJmNiIvPgogIDx0ZXh0IHg9
IjIwMCIgeT0iMzE5IiBmb250LWZhbWlseT0iQXJpYWwsIHNhbnMtc2VyaWYiIGZvbnQtc2l6ZT0iM
TAiIGZpbGw9IiMzMzMiPkZ1bGwtc3RhY2s8L3RleHQ+CiAgPGNpcmNsZSBjeD0iMjgwIiBjeT0iMz
E1IiByPSI4IiBmaWxsPSIjZjU5ZTBiIi8+CiAgPHRleHQgeD0iMjk1IiB5PSIzMTkiIGZvbnQtZmF
taWx5PSJBcmlhbCwgc2Fucy1zZXJpZiIgZm9udC1zaXplPSIxMCIgZmlsbD0iIzMzMyI+WmVybyBj
b25maWc8L3RleHQ+CiAgPGNpcmNsZSBjeD0iMzg1IiBjeT0iMzE1IiByPSI4IiBmaWxsPSIjOGI1Y
2Y2Ii8+CiAgPHRleHQgeD0iNDAwIiB5PSIzMTkiIGZvbnQtZmFtaWx5PSJBcmlhbCwgc2Fucy1zZX
JpZiIgZm9udC1zaXplPSIxMCIgZmlsbD0iIzMzMyI+RGlzdHJpYnV0ZWQgdHJhY2luZzwvdGV4dD4
KICA8Y2lyY2xlIGN4PSI1MTUiIGN5PSIzMTUiIHI9IjgiIGZpbGw9IiNlZjQ0NDQiLz4KICA8dGV4
dCB4PSI1MzAiIHk9IjMxOSIgZm9udC1mYW1pbHk9IkFyaWFsLCBzYW5zLXNlcmlmIiBmb250LXNpe
mU9IjEwIiBmaWxsPSIjMzMzIj5Mb2cgY29sbGVjdGlvbjwvdGV4dD4KPC9zdmc+Cg==)
```

```


## 2. Deployment Strategy

### Phased Rollout (Recommended)

Don't deploy everywhere at once. Use a phased approach:

| Phase | Scope | Purpose |
|-------|-------|---------|
| **Pilot** | 2-5 non-critical hosts | Validate installation, network,
discovery |
| **Expand** | One application tier | Test service detection, tracing |
| **Production** | Full environment | Complete coverage |

### Deployment Methods by Environment

| Environment | Recommended Method |
|-------------|-------------------|
| **Bare metal/VMs** | Direct install or package manager |
| **Kubernetes** | Dynatrace Operator |
| **OpenShift** | Dynatrace Operator |
| **AWS ECS** | Task definition sidecar |
| **Azure** | VM Extension or AKS Operator |
| **GCP** | Direct install or GKE Operator |

### Network Requirements

OneAgent needs outbound HTTPS (443) to:
- `*.apps.dynatrace.com` (SaaS)

If direct access isn't possible, OneAgents connect through **ActiveGate**
```

(see ONBRD-03).

## 3. Generating a Deployment Token

You need an API token with installer download permissions.

Location: Account Management → Access tokens → Generate new token

### Required Scope

Scope	API Name	Purpose
PaaS integration - Installer download	`InstallerDownload`	Download OneAgent installer

### Token Naming Convention

Use descriptive names:
- `prod-oneagent-deployment`
- `dev-installer-token`
- `k8s-operator-token`

> **Important:** Copy the token immediately after creation. It won't be shown again.

## 4. Installation Methods

### Linux (Direct Download)

```bash
# Download the installer
wget -O Dynatrace-OneAgent.sh \
  "https://{tenant-id}.apps.dynatrace.com/api/v1/deployment/installer/agent/unix/default/latest?Api-Token={token}&arch=x86&flavor=default"

# Run the installer
sudo /bin/sh Dynatrace-OneAgent.sh
```

### Linux (One-liner)

```bash
wget -O Dynatrace-OneAgent.sh "https://{tenant-id}.apps.dynatrace.com/api/v1/deployment/installer/agent/unix/default/latest?Api-Token={token}" && sudo /bin/sh Dynatrace-OneAgent.sh
```

~~~
### Windows (PowerShell)

```powershell
# Download the installer
Invoke-WebRequest -Uri "https://{tenant-id}.apps.dynatrace.com/api/v1/deployment/installer/agent/windows/default/latest?Api-Token={token}" -OutFile Dynatrace-OneAgent.exe

# Run the installer
.\Dynatrace-OneAgent.exe
```

### Configuration Management

For Ansible, Puppet, Chef, or other tools, see:
- [Ansible Collection](https://docs.dynatrace.com/docs/setup-and-configuration/dynatrace-oneagent/installation-and-operation/linux/installation/install-oneagent-on-linux#expand--oneagent-deployment-automation-with-ansible--2)
- [Puppet Module](https://forge.puppet.com/modules/dynatrace/dynatrace_oneagent)
- [Chef Cookbook](https://supermarket.chef.io/cookbooks/dynatrace)

## 5. Kubernetes Deployment

For Kubernetes environments, use the Dynatrace Operator.

### Prerequisites

- `kubectl` access to cluster
- API token with required scopes
- Cluster admin permissions

### Required Token Scopes for Kubernetes

| Scope | Purpose |
|-------|---------|
| `InstallerDownload` | Download OneAgent |
| `entities.read` | Read topology |
| `settings.read` | Read configuration |
| `settings.write` | Write configuration |
| `DataExport` | Export data (optional) |

### Installation Steps

1. **Navigate to Kubernetes monitoring**
   - Infrastructure → Kubernetes → Add cluster
~~~

~~~
2. **Follow the wizard**
   - Select your distribution
   - Choose observability options
   - Download dynakube.yaml

3. **Apply to cluster**
   ```bash
   kubectl apply -f https://github.com/Dynatrace/dynatrace-operator/releases/download/v1.3.2/kubernetes.yaml
   kubectl apply -f dynakube.yaml
   ```

4. **Verify deployment**
   ```bash
   kubectl get pods -n dynatrace
   ```

## 6. Verifying Deployment

After installation, verify OneAgent is reporting data.

```dql
// Check all hosts with OneAgent
fetch dt.entity.host
| fields entity.name, state, monitoringMode
| filter state == "RUNNING"
| sort entity.name
| limit 50
```

```dql
// Check hosts by monitoring state
fetch dt.entity.host
| summarize host_count = count(), by: {state, monitoringMode}
| sort host_count desc
```

```dql
// Check hosts by monitoring state - useful for verifying deployment
fetch dt.entity.host
| fields entity.name, state, monitoringMode
| summarize host_count = count(), by: {state}
| sort host_count desc
```

```dql
// Check discovered services
fetch dt.entity.service
~~~

~~~
| fields entity.name, serviceType
| summarize service_count = count(), by: {serviceType}
| sort service_count desc
```

```dql
// Check for processes discovered
fetch dt.entity.process_group
| fields entity.name
| sort entity.name
| limit 50
```

### Host Verification Commands

**Linux:**
```bash
# Check OneAgent status
sudo systemctl status oneagent

# Check connection to Dynatrace
sudo /opt/dynatrace/oneagent/agent/tools/oneagent-connection-check

# Check OneAgent logs
sudo tail -100 /var/log/dynatrace/oneagent/oneagent.log
```

**Windows:**
```powershell
# Check service status
Get-Service -Name "Dynatrace OneAgent"

# Check connection
& "C:\Program Files\dynatrace\oneagent\agent\tools\oneagent-connection-check.exe"
```

## 7. Troubleshooting

### Common Issues

| Issue | Cause | Solution |
|-------|-------|----------|
| **Host not appearing** | Network blocked | Check firewall rules for 443 |
| **Host showing but no data** | Agent not running | Restart OneAgent service |
| **Services not discovered** | Processes not restarted | Restart monitored applications |
~~~

~~~
| **Old OneAgent version** | Auto-update disabled | Enable auto-update or manual upgrade |
| **Connection errors** | Proxy required | Configure ActiveGate or proxy |

### Network Verification

```bash
# Test connectivity to Dynatrace
curl -v https://{tenant-id}.apps.dynatrace.com/api/v1/time

# Check DNS resolution
nslookup {tenant-id}.apps.dynatrace.com

# Test with OpenSSL
openssl s_client -connect {tenant-id}.apps.dynatrace.com:443
```

### Process Restart Requirement

OneAgent injects into running processes. After initial installation:

- **New processes** - Automatically monitored
- **Existing processes** - Restart required for full monitoring

For deep code-level visibility, restart:
- Application servers (Tomcat, JBoss, etc.)
- Web servers (Apache, Nginx, IIS)
- .NET/Java applications
- Node.js applications

## 8. Next Steps

With OneAgent deployed and verified:

1. **ONBRD-06: Organizing Your Environment** - Set up tags, segments, and naming conventions
2. Wait 15-30 minutes for full topology discovery
3. Check Smartscape for service dependencies
4. Plan broader rollout based on pilot results

### Deployment Checklist

- [ ] Pilot hosts deployed (2-5 hosts)
- [ ] Hosts appearing in Dynatrace
- [ ] OneAgent status "RUNNING" on all hosts
- [ ] Services being discovered
- [ ] Connection check passing
- [ ] Smartscape showing topology
~~~

```
---

## Summary

In this notebook, you learned:

- What OneAgent does and how it works
- Phased deployment strategy
- How to generate deployment tokens
- Installation methods for Linux, Windows, and Kubernetes
- How to verify successful deployment
- Common troubleshooting steps

---

## References

- [OneAgent Overview](https://docs.dynatrace.com/docs/setup-and-configuration/dynatrace-oneagent)
- [Linux Installation](https://docs.dynatrace.com/docs/setup-and-configuration/dynatrace-oneagent/installation-and-operation/linux/installation/install-oneagent-on-linux)
- [Windows Installation](https://docs.dynatrace.com/docs/setup-and-configuration/dynatrace-oneagent/installation-and-operation/windows/installation/install-oneagent-on-windows)
- [Kubernetes Operator](https://docs.dynatrace.com/docs/ingest-from/setup-on-k8s)
- [Deployment API](https://docs.dynatrace.com/docs/dynatrace-api/environment-api/deployment)
```