The DevOps Lifecycle

Security's Role in
freedom to focus on the mission.
secure releases, greater interoperability and the
undergoing digital transformation: faster, more
advantage for government agencies who are
security—at scale. The result delivers powerful
of continuous improvement and integrated
DevSecOps is a philosophy, movement, and culture
of development. In order to take full advantage of a
an isolated team—tacked on in the final stage of
Security used to be the exclusive responsibility of
Security is a shared
responsibility, integrated
processes are open, collaborative,
materials, decisions, and
optimal policies that provide a single
migrate applications for better
improve your current processes,
automation technology that can
A simple, enterprise IT
DevOps is for everyone
while DevOps specifically refers to development, integration, and
agile approaches to work.
DevOps relies on a culture of collaboration that aligns with open
the practice is most successful when spread throughout the agency.
While DevOps specifically refers to development and operations,
Agile processes help to facilitate an open culture.
With the right leadership and incentives in place, your development
source principles and transparent, agile approaches to work.
Open organizations are communal by
inclusivity where everyone is heard,
the container build process.
Conservatism is isolation, and checks throughout
Red Hat® OpenShift® delivers
built-in security for
container-based applications,
including role-based access
controls, Security-Enhanced
Linux (SELinux)-enabled
security and meet your DevOps
Select the right tools to integrate
security into your workflow
and keep the team focused on
Identify the security controls
necessary within a given app
Shift security left, as early as
the planning stage
Prioritize speed to market for
different apps
Treat your infrastructure like dev teams treat code
Treat your infrastructure like dev teams treat code
Automation: the process of continuous delivery
A process of continuous delivery
full lifecycle support for software delivered under the umbrella of continuous delivery
is automation.
1. Use an open, modular approach to automation
2. Selecting tools that support your processes is critical
3. Integrate security measures with minimal disruption to operations.
4. Foster closer collaboration between commonly isolated teams.
5. Keep up with innovative technologies like containers and microservices.
7. Allows environments to continuously scale with ease.
8. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
9. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
10. Foster closer collaboration between commonly isolated teams.
11. Keep up with innovative technologies like containers and microservices.
12. Integrate security measures with minimal disruption to operations.
14. Allows environments to continuously scale with ease.
15. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
16. Foster closer collaboration between commonly isolated teams.
17. Keep up with innovative technologies like containers and microservices.
18. Integrate security measures with minimal disruption to operations.
20. Allows environments to continuously scale with ease.

The DevSecOps Tools You Need:

Nonprofit Episode Application Platform
A non-profit episode application platform that
is fully managed and run by Red Hat
Launchpad application management
for DevSecOps
OpenShift Container Platform
OpenShift Container Platform is an
open source Kubernetes distribution
that is used to deploy and manage
containers in a Kubernetes cluster.
OpenShift is a cloud-native platform
that provides a comprehensive
solution for building, deploying,
and managing containerized
applications.
OpenShift is an open source Kubernetes
distribution that is used to deploy and
manage containers in a Kubernetes
cluster.
OpenShift is a cloud-native platform
that provides a comprehensive
solution for building, deploying,
and managing containerized
applications.
OpenShift is an open source Kubernetes
distribution that is used to deploy and
manage containers in a Kubernetes
cluster.
OpenShift is a cloud-native platform
that provides a comprehensive
solution for building, deploying,
and managing containerized
applications.

To learn more about Red Hat technologies enabling and driving DevSecOps, please contact:

Red Hat Solutions for Government
Email: RedHat@carahsoft.com
Toll-Free: (877)-RHAT-GOV
Main: (703)-871-8570
Fax: (703)-871-8505

Powers Public Sector Innovation
Continuous improvement and
There is no one way
that works for everyone:
Your team—likely
across your organization.
language for DevOps practices
optimization, and provide a single
migrate applications for better
improve your current processes,
automation technology that can
A simple, enterprise IT
DevOps is for everyone
while DevOps specifically refers to development, integration, and
agile approaches to work.
DevOps relies on a culture of collaboration that aligns with open
the practice is most successful when spread throughout the agency.
While DevOps specifically refers to development and operations,
Agile processes help to facilitate an open culture.
With the right leadership and incentives in place, your development
source principles and transparent, agile approaches to work.
Open organizations are communal by
inclusivity where everyone is heard,
the container build process.
Conservatism is isolation, and checks throughout
Red Hat® OpenShift® delivers
built-in security for
container-based applications,
including role-based access
controls, Security-Enhanced
Linux (SELinux)-enabled
security and meet your DevOps
Select the right tools to integrate
security into your workflow
and keep the team focused on
Identify the security controls
necessary within a given app
Shift security left, as early as
the planning stage
Prioritize speed to market for
different apps
Treat your infrastructure like dev teams treat code
Treat your infrastructure like dev teams treat code
Automation: the process of continuous delivery
A process of continuous delivery
full lifecycle support for software delivered under the umbrella of continuous delivery
is automation.
1. Use an open, modular approach to automation
2. Selecting tools that support your processes is critical
3. Integrate security measures with minimal disruption to operations.
4. Foster closer collaboration between commonly isolated teams.
5. Keep up with innovative technologies like containers and microservices.
7. Allows environments to continuously scale with ease.
8. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
9. Foster closer collaboration between commonly isolated teams.
10. Keep up with innovative technologies like containers and microservices.
11. Integrate security measures with minimal disruption to operations.
13. Allows environments to continuously scale with ease.
14. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
15. Foster closer collaboration between commonly isolated teams.
16. Keep up with innovative technologies like containers and microservices.
17. Integrate security measures with minimal disruption to operations.
18. Maintain short and frequent development cycles.
19. Allows environments to continuously scale with ease.
20. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
21. Foster closer collaboration between commonly isolated teams.
22. Keep up with innovative technologies like containers and microservices.
23. Integrate security measures with minimal disruption to operations.
25. Allows environments to continuously scale with ease.
26. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
27. Foster closer collaboration between commonly isolated teams.
28. Keep up with innovative technologies like containers and microservices.
29. Integrate security measures with minimal disruption to operations.
30. Maintain short and frequent development cycles.
31. Allows environments to continuously scale with ease.
32. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
33. Foster closer collaboration between commonly isolated teams.
34. Keep up with innovative technologies like containers and microservices.
35. Integrate security measures with minimal disruption to operations.
36. Maintain short and frequent development cycles.
37. Allows environments to continuously scale with ease.
38. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
39. Foster closer collaboration between commonly isolated teams.
40. Keep up with innovative technologies like containers and microservices.
41. Integrate security measures with minimal disruption to operations.
42. Maintain short and frequent development cycles.
43. Allows environments to continuously scale with ease.
44. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
45. Foster closer collaboration between commonly isolated teams.
46. Keep up with innovative technologies like containers and microservices.
47. Integrate security measures with minimal disruption to operations.
49. Allows environments to continuously scale with ease.
50. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
51. Foster closer collaboration between commonly isolated teams.
52. Keep up with innovative technologies like containers and microservices.
53. Integrate security measures with minimal disruption to operations.
54. Maintain short and frequent development cycles.
55. Allows environments to continuously scale with ease.
56. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
57. Foster closer collaboration between commonly isolated teams.
58. Keep up with innovative technologies like containers and microservices.
59. Integrate security measures with minimal disruption to operations.
60. Maintain short and frequent development cycles.
61. Allows environments to continuously scale with ease.
62. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
63. Foster closer collaboration between commonly isolated teams.
64. Keep up with innovative technologies like containers and microservices.
65. Integrate security measures with minimal disruption to operations.
67. Allows environments to continuously scale with ease.
68. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
69. Foster closer collaboration between commonly isolated teams.
70. Keep up with innovative technologies like containers and microservices.
71. Integrate security measures with minimal disruption to operations.
72. Maintain short and frequent development cycles.
73. Allows environments to continuously scale with ease.
74. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
75. Foster closer collaboration between commonly isolated teams.
76. Keep up with innovative technologies like containers and microservices.
77. Integrate security measures with minimal disruption to operations.
78. Maintain short and frequent development cycles.
79. Allows environments to continuously scale with ease.
80. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
81. Foster closer collaboration between commonly isolated teams.
82. Keep up with innovative technologies like containers and microservices.
83. Integrate security measures with minimal disruption to operations.
84. Maintain short and frequent development cycles.
85. Allows environments to continuously scale with ease.
86. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
87. Foster closer collaboration between commonly isolated teams.
88. Keep up with innovative technologies like containers and microservices.
89. Integrate security measures with minimal disruption to operations.
90. Maintain short and frequent development cycles.
91. Allows environments to continuously scale with ease.
92. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
93. Foster closer collaboration between commonly isolated teams.
94. Keep up with innovative technologies like containers and microservices.
95. Integrate security measures with minimal disruption to operations.
96. Maintain short and frequent development cycles.
97. Allows environments to continuously scale with ease.
98. Enables an infrastructure to withstand the constant code
changes that come with DevOps.
99. Foster closer collaboration between commonly isolated teams.
100. Keep up with innovative technologies like containers and microservices.
101. Integrate security measures with minimal disruption to operations.
102. Maintain short and frequent development cycles.
103. Allows environments to continuously scale with ease.