



PRACTITIONER

You've mastered CI - now what? How to actually do CD on k8s

Kevin Ng

SOLUTION ARCHITECT





Kevin Ng

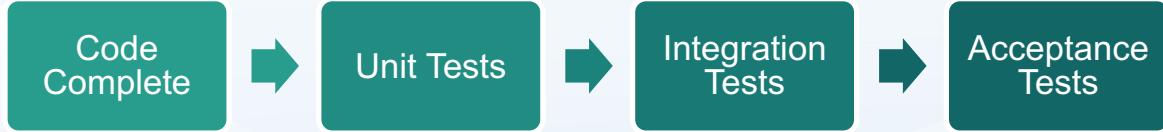
Passionate about helping
people identify issues and find
solutions to overcome them

Takeaways

- How to update an application without breaking production
- How to avoid introducing vulnerabilities
- How to handle security when automatically deploying changes
- The right way to update a running application on Kubernetes

CI/CD

Continuous Integration



Continuous Delivery/Deployment



Continuous Release

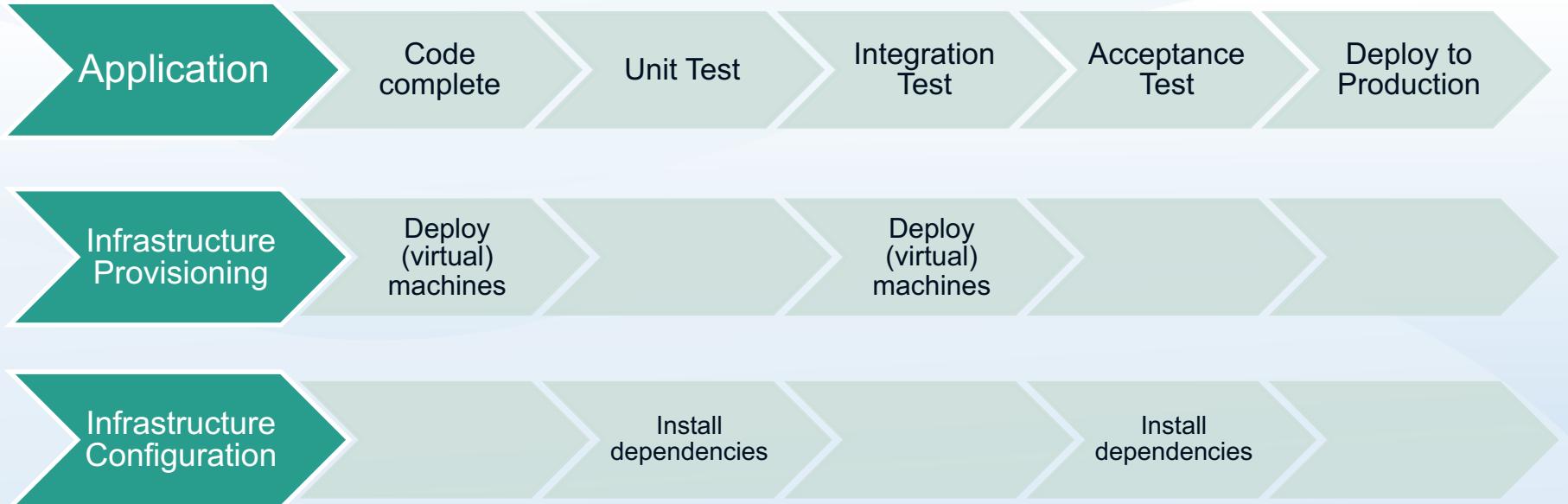


How to update an application without breaking production

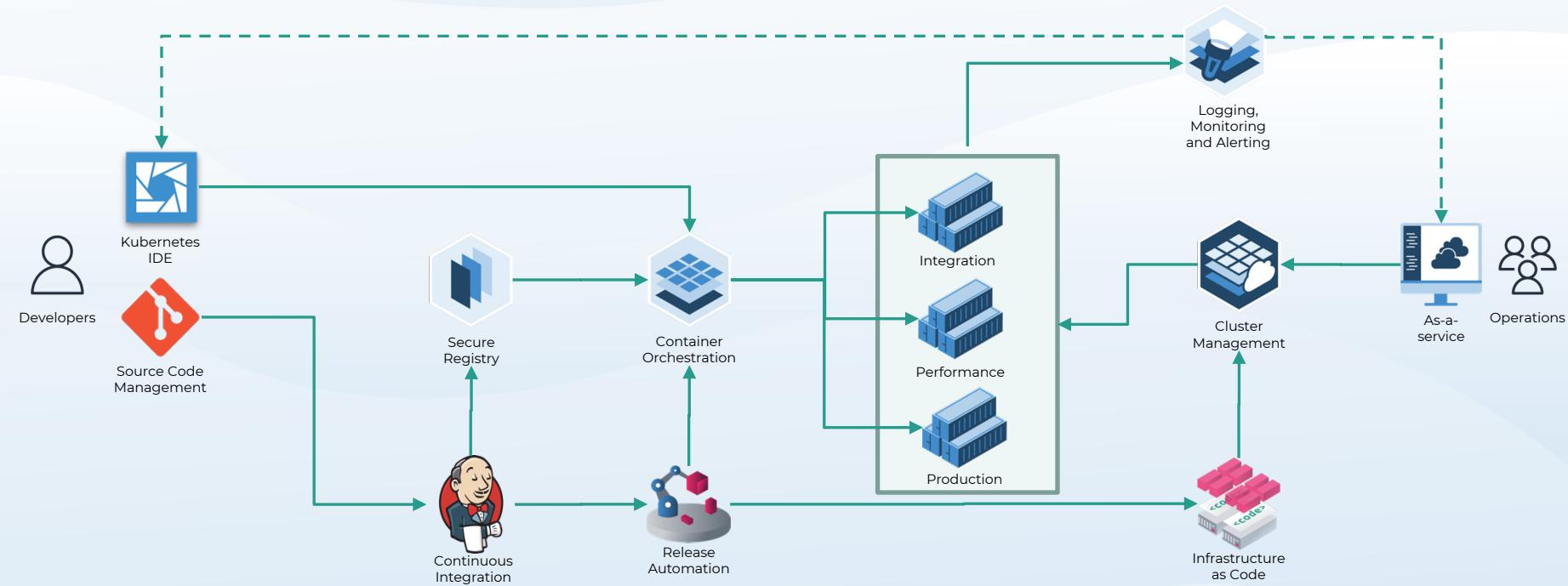
Thoroughly test throughout the pipeline



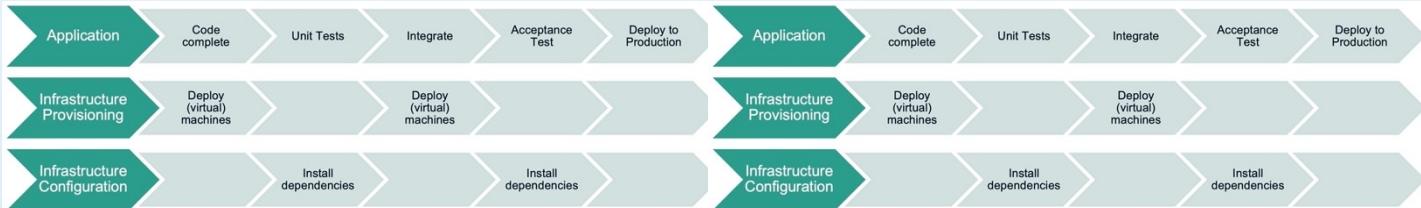
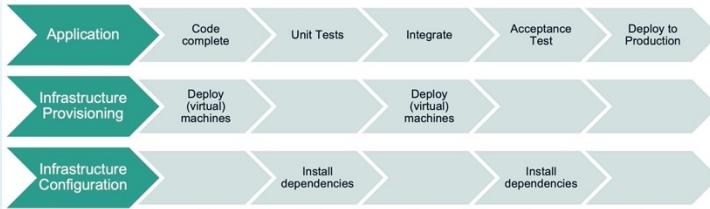
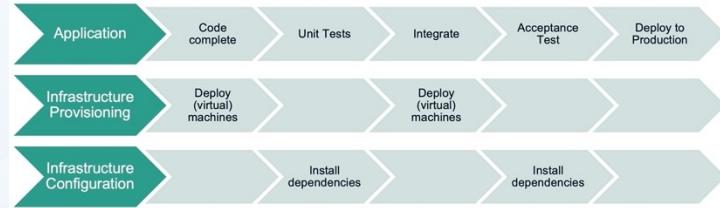
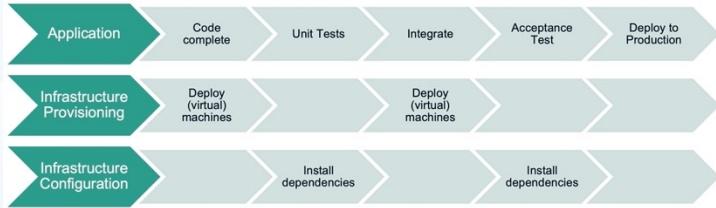
Application deployment process



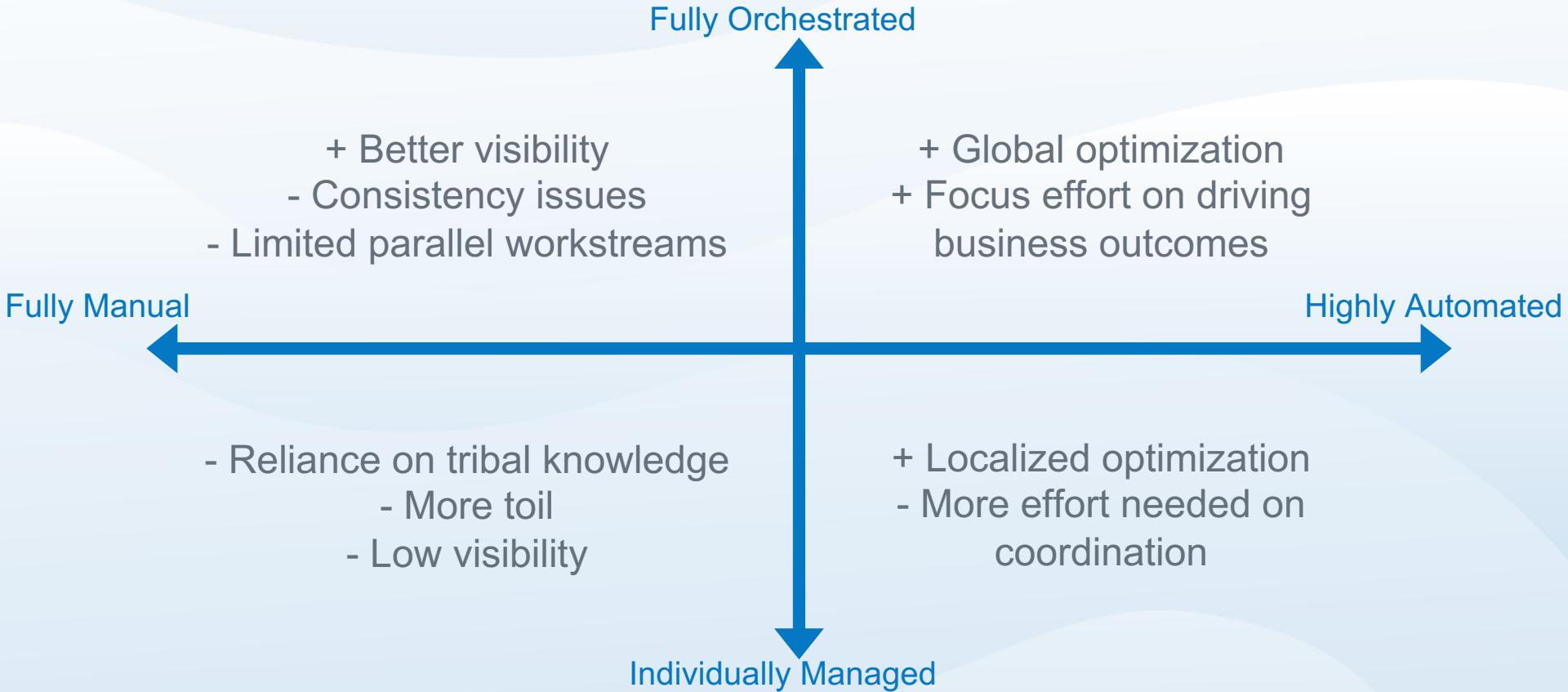
Automate the deployment pipeline



Application deployment process across teams



Automation and Orchestration



How to avoid introducing vulnerabilities

Use approved base images



Thoroughly scan images

Repositories / kng-prod / simple-nginx / 2021.08.20.174424 / Components

kng-prod / simple-nginx: 2021.08.20.174424

linux / amd64	9ecf99c1584c	10.99 MB	Pushed 4 days ago by jenkins	Out Of Date	1 critical	6 high	6 medium	2 low	All layers already scanned	Delete	Promote	Scan
Layers	Components											
nginx												
● 1.19.10-r1			Critical 1 High 0 Medium 0 Low 0									
curl				curl								
● 7.74.0-r1			Critical 0 High 3 Medium 4 Low 2	Version: 7.74.0-r1								
libxml2				License: MIT								
● 2.9.10-r6			Critical 0 High 0 Medium 1 Low 0	Vulnerabilities		Severity	Description					
gdlib				CVE-2021-22922	7.8		When curl is instructed to download content using the metalink feature, the contents is verified against a hash provided in the metalink XML file. The metalink XML file points out to the client how to get the same content from a set of different URLs, potentially hosted by different servers and the client can then download the file from one or several of them. In a serial or parallel manner. If one of the servers hosting the contents has been breached and the contents of the specific file on that server is replaced with a modified payload, curl should detect this when the hash of the file mismatches after a completed download. It should remove the contents and instead try getting the contents from another URL. This is not done, and instead such a hash mismatch is only mentioned in text and the potentially malicious content is kept in the file on disk.	Show layers affected				
libxslt				CVE-2021-22924	7.4		libcurl keeps previously used connections in a connection pool for subsequent transfers to reuse, if one of them matches the setup. Due to errors in the logic, the config matching function did not take 'issuercert' into account and it compared the involved paths *case insensitively*, which could lead to libcurl reusing wrong connections. File paths are, or can be, case sensitive on many systems but not all, and can even vary depending on used file systems. The comparison also didn't include the 'issuer cert' which a transfer can set to qualify how to	Show layers affected				
				File paths								
				usr/lib/libcurl.so.4.7.0	usr/bin/curl							

How to handle security when automatically deploying changes

Sign deployment images

[Repositories](#) / kng-prod / simple-nginx / Tags

kng-prod / simple-nginx

Info	Permissions	Tags	Webhooks	Promotions	Pruning	Mirrors	Settings	Activity	
		Image	Type	Digest	Size	Signed	Last Pushed	Vulnerabilities	
<input type="checkbox"/>		2021.08.20.174424	linux amd64	9ecf99c1584c	11.52 MB	Signed	4 days ago by jenkins	Critical 1 High 6 Medium 6 Low 2	View details
<input type="checkbox"/>		2021.08.17.170400	linux amd64	9ecf99c1584c	11.52 MB	Signed	7 days ago by jenkins	Critical 1 High 6 Medium 6 Low 2	View details
 <	 >								

Promote, don't rebuild

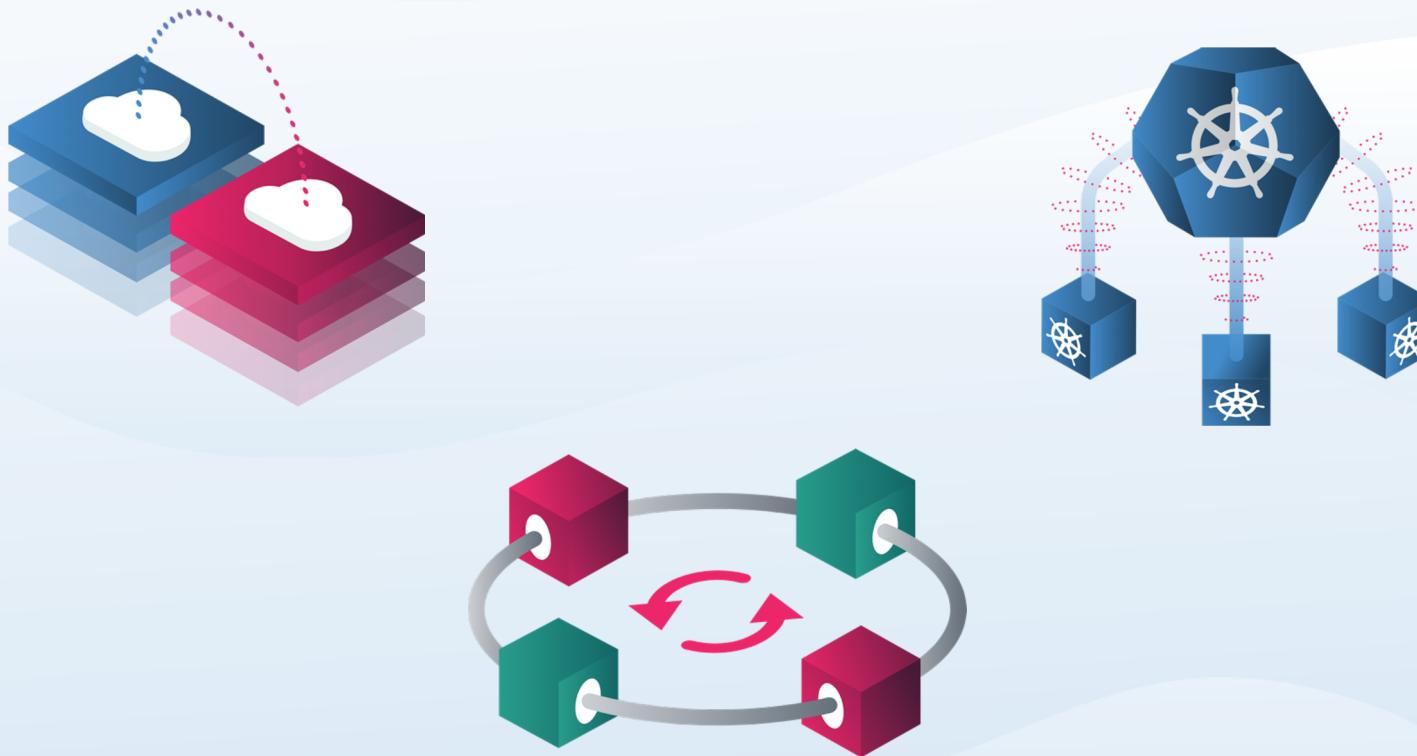
The screenshot shows a container registry interface for a repository named "kng-dev / simple-nginx". The main card displays the image details: tag "2021.08.20.174424", pushed 4 days ago, size 10.99 MB, and a security summary with 6 medium and 2 low vulnerabilities. A "Promote" button is visible.

A modal dialog titled "Promote Tagged Image" is open, prompting the user to select a target repository and tag name. The "TARGET REPOSITORY" dropdown is set to "kng-prod". The "Tag Name In Target" input field contains "simple-nginx". A dropdown menu lists two options: "pacman-nodejs" and "simple-nginx".

At the bottom of the dialog are "Cancel" and "Promote" buttons.

The right way to update a running application on Kubernetes

Utilize deployment patterns



Increase environment observability

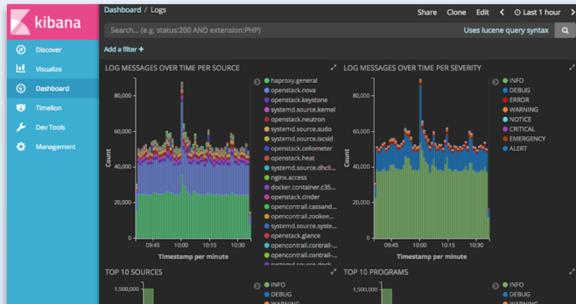
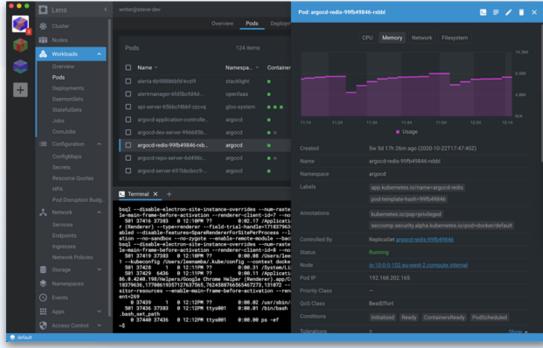


The Alerta interface displays a list of critical alerts across different environments and services. Key details from the table:

Severity	Status	Last Received	Dept.	Environment	Service	Resource	Event	Value	Unit
critical	open	Thu 19 Aug 06:27	3547	Development	node	cmp118/remote_agent/Epel6stack	NewlyOpen/Received/0/critical	16800MB of disk space on the cmp118 node (> 95.0%) is used.	MB
critical	open	Thu 19 Aug 06:27	3547	Development	node	cmp122/remote_agent/Epel6stack	NewlyOpen/Memory/0/critical	25445MB of RAM on the cmp122 node (> 95.0%) is used.	MB
critical	open	Thu 19 Aug 06:27	3547	Development	node	cmp06/remote_agent/Epel6stack	NewlyOpen/Memory/0/critical	25445MB of RAM on the cmp06 node (> 95.0%) is used.	MB
warn	open	Thu 19 Aug 06:27	420	Development	node	cmp106/remote_agent/Epel6stack	NewlyOpen/NetworkUtilization	25000MB of RAM on the cmp106 node (> 95.0%) is used.	MB
warn	open	Thu 19 Aug 06:27	3548	Development	node	cmp106/remote_agent/Epel6stack	NewlyOpen/NetworkUtilization	15400MB of disk space on the cmp106 node (> 95.0%) is used.	MB
warn	open	Thu 19 Aug 06:27	3548	Development	node	cmp06/remote_agent/Epel6stack	NewlyOpen/NetworkUtilization	25445MB of RAM on the cmp06 node (> 95.0%) is used.	MB
warn	open	Wed 18 Aug 15:56	5	Development	elasticsearch	logstash	Elasticsearch/AvailableMemory	The Elasticsearch '172.17.4.6:9200' instance uses 25445MB of memory (95.0%) of the available 26500MB on the right node for 5 minutes.	MB
warn	open	Thu 19 Aug 06:27	3549	Development	system	cmpr01/Logstash	SystemMemoryWarning	25000MB of RAM on the cmpr01 node (> 95.0%) is used.	MB
warn	open	Thu 19 Aug 06:27	3549	Development	system	prod/Logstash	SystemMemoryWarning	The disk partition mounted on the prod node is 85.2223513548494%, full for 2 minutes.	%
warn	open	Thu 19 Aug 06:28	3550	Development	system	mon/Logstash	SystemLoad/ConfigMemoryWarning	The system load per CPU on the mon01 node is 3.533 for 6 minutes.	None



Promote information sharing between teams



Recap

Recap: updating applications safely

- Test the application and pipeline thoroughly to ensure reliability
- Automate the deployment pipeline to reduce manual errors and allow for process improvement
- Orchestrate all activities to account for dependencies and bottlenecks

Recap: avoid introducing vulnerabilities

- Use approved base images to avoid malicious code
- Perform thorough scans on images to proactively identify vulnerabilities

Recap: securely deploy images

- Ensure deployment images are signed for authenticity
- Promote images throughout environments to avoid introducing vulnerabilities

Recap: updating applications with Kubernetes

- Utilize deployment patterns to control release effects
- Increase environment observability for rapid response
- Promote information sharing between teams to facilitate continuous improvement



Thank you!

Kevin Ng

LinkedIn: <https://www.linkedin.com/in/kevinkng/>

Twitter: @kevng9

Email: kng@mirantis.com

Find out more at mirantis.com or k8slens.dev