

Skills modernization for the Virtual Infrastructure Admin

Take the first step using VEBA

Patrick Kremer
Staff Cloud Solution Architect, VMC on AWS

Personal Blog: <http://www.patrickkremer.com>
Twitter: @KremerPatrick

September 2020

About Me

VMware Employee since
2014 – Presales SE

Started working with
VMware in 2006 - ESX 3.0

Primarily a sysadmin

Limited programming
background

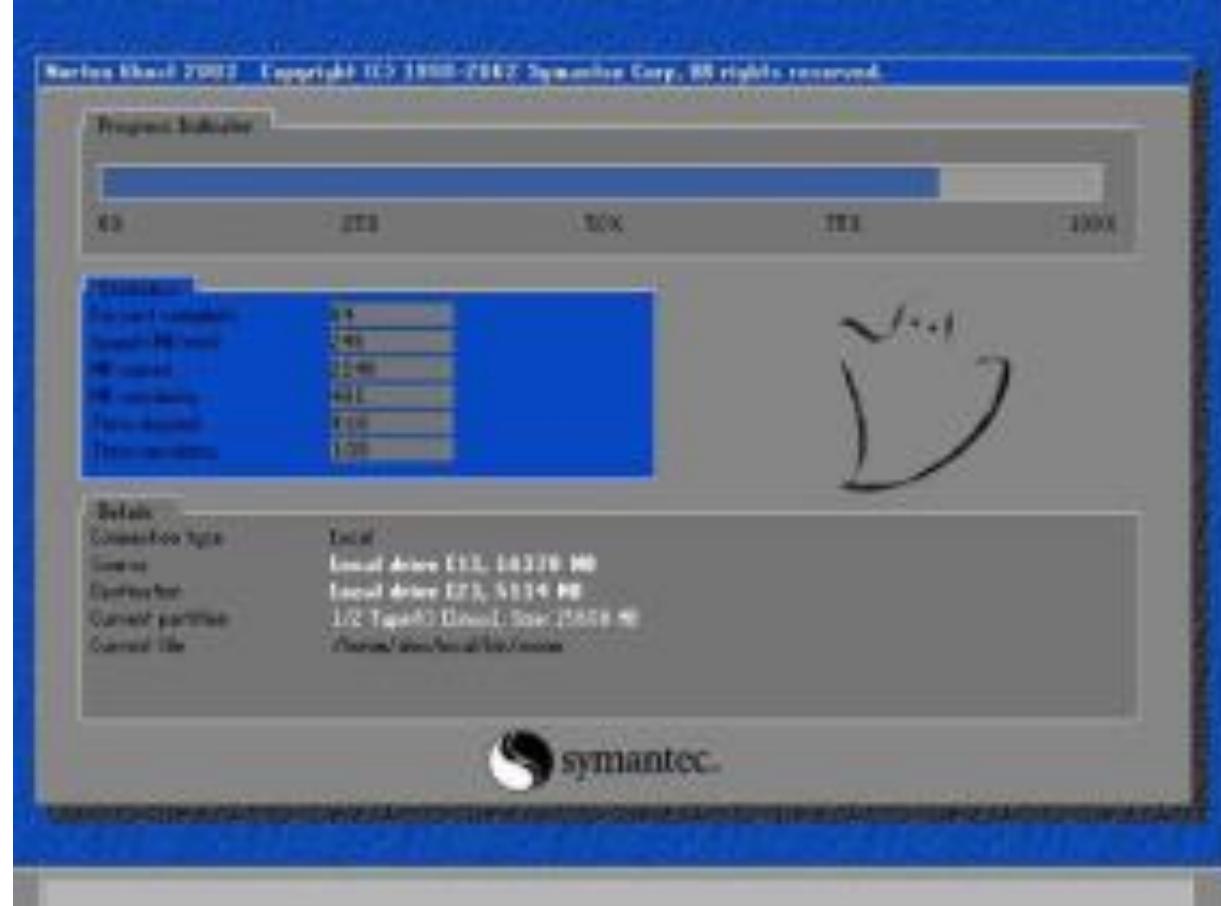
Currently a VMware Cloud
Solution Architect - VMC



Why Modernize your Skills?

Obsolescence

Having no development skills will soon be equal to having no virtualization skills today.



In the beginning



THE
BEGINNING



Microsoft® Visual Basic® for Applications

Customers

Customers

ID	Company	First Name	Last Name	E-mail Address
1	Company A	Anna	Bedecs	
2	Company B	Antonio	Gratacos Solso	
3	Company C	Thomas	Axen	
4	Company D	Christina	Lee	
5	Company E	Martin	O'Donnell	
6	Company F	Francisco	Pérez-Olaeta	
7	Company G	Ming-Yang	Xie	
8	Company H	Elizabeth	Andersen	

City:
State/P:
ZIP/Pos:
Country:
Web Pa:
Notes:
Attachments:

Record: 1 of 29 | No Filter | Search |

Since 2006

All system administration, all VMs, all 3-tier apps, all the time



Three Tier Apps



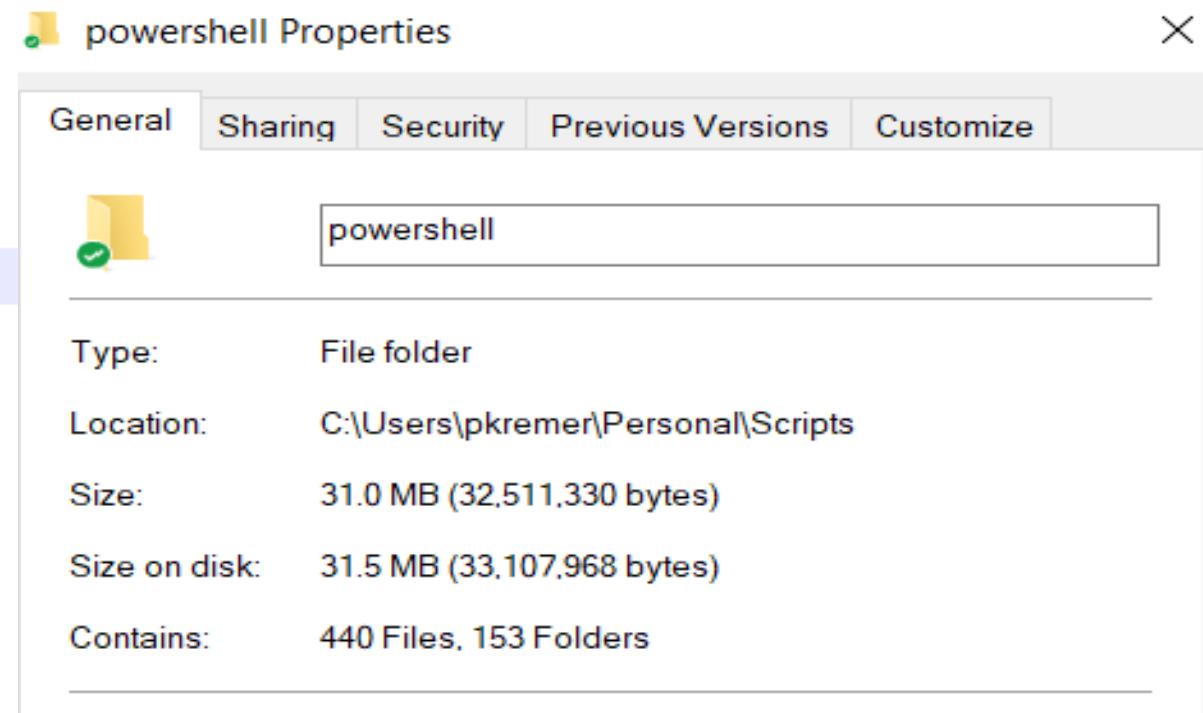
Virtual Machines



vSphere

Programming = Scripting

```
1 $AllClusters = Get-Cluster | Where {$_.Name -ne "LabManager" -and $_.Name -ne "Double Take Windows 2000 ONLY" } | Sort-Object "Name"
2 ForEach( $Cluster in $AllClusters)
3 {
4     $ClusterName = $Cluster.Name
5     $AllVMs = get-cluster $ClusterName | Get-VM | Where { $_.Guest.State -ne "NotRunning" -And $_.Description -like "*2015*" } | Sort-Object
6         Name
7     ForEach ($VM in $AllVMs )
8     {
9         Write-Host $VM.Name
10        Write-Host $VM.Description
11    }
```



2013



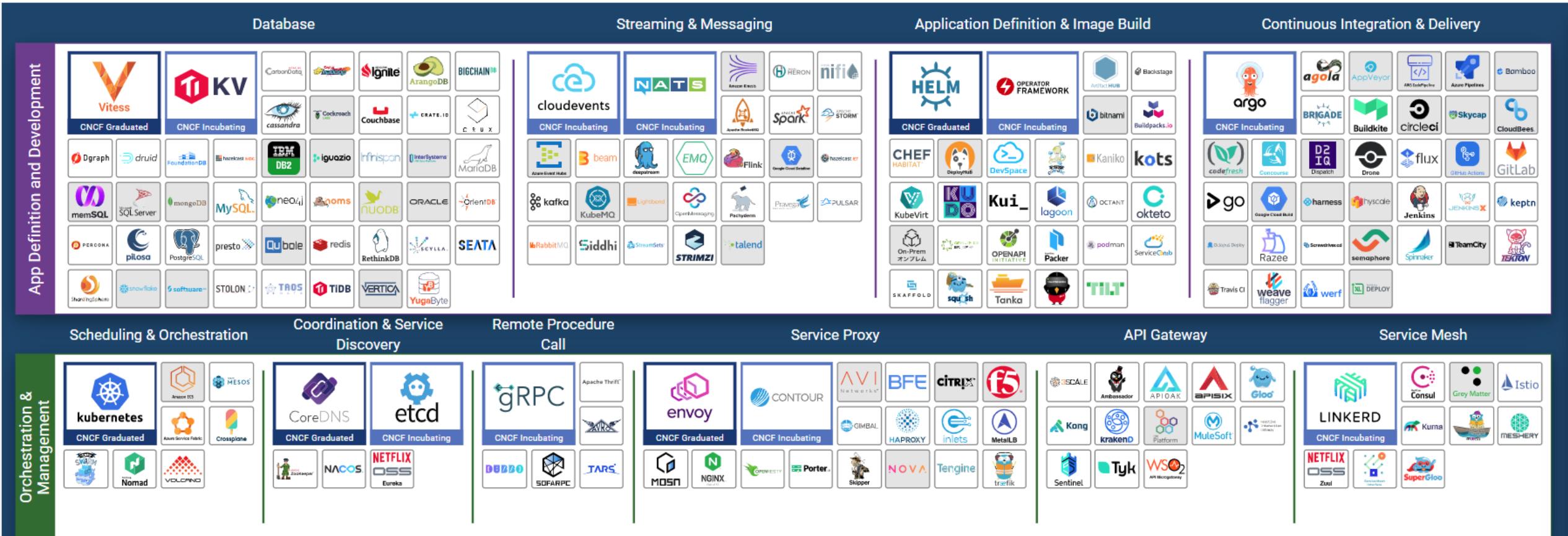
VMware

NSX

2020

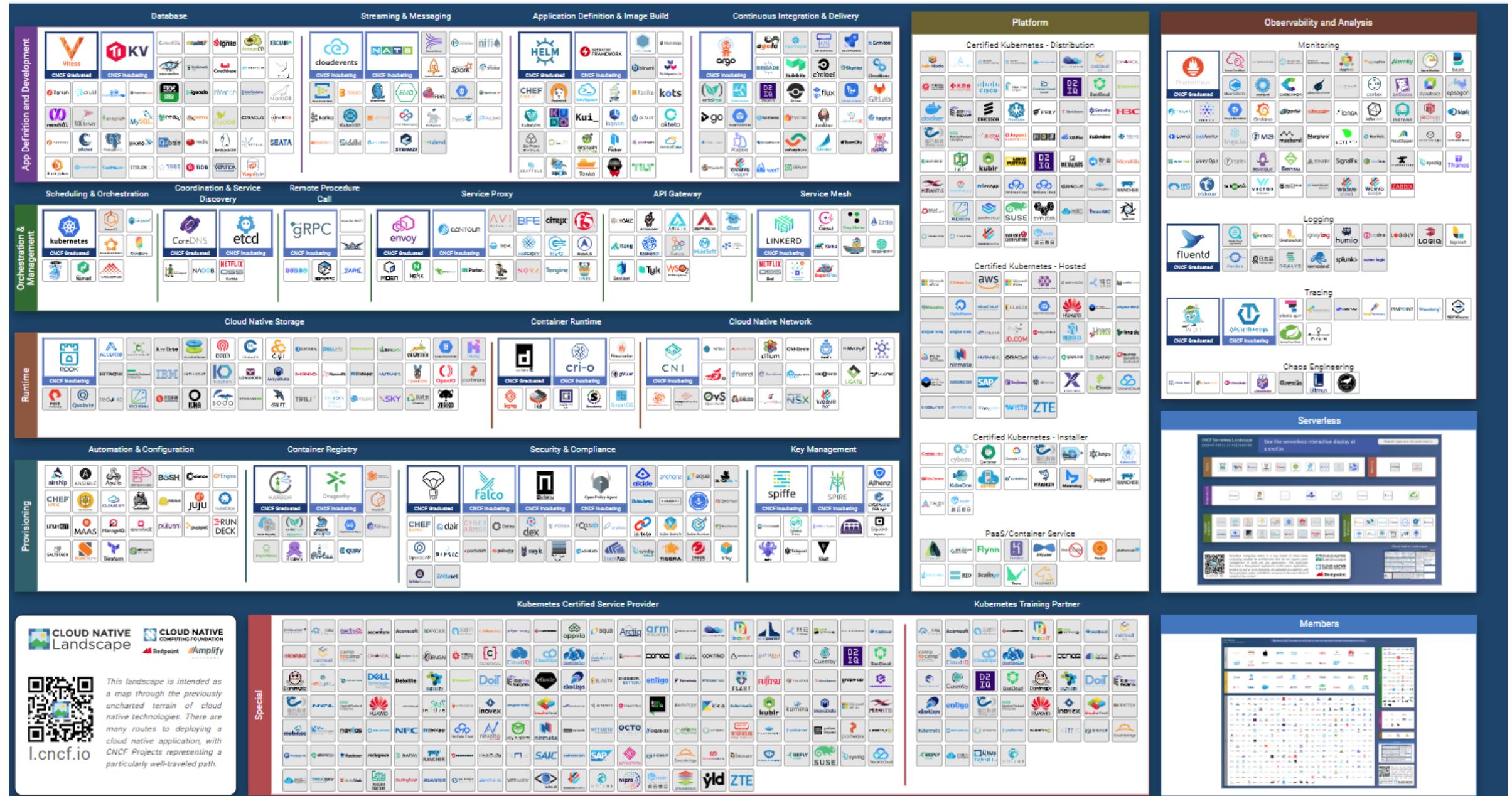
gitdevops
containers
microservices
Kubernetes
Github
CI/CD
agile
Docker

Cloud Native Landscape

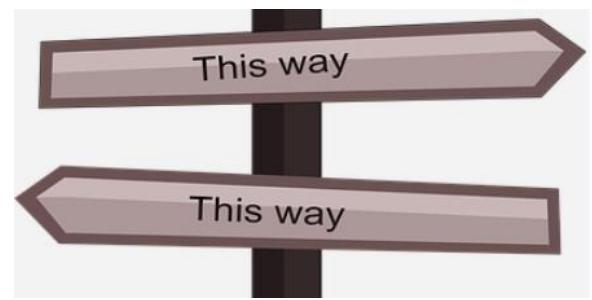


Cloud Native Landscape

Source: <https://landscape.cncf.io/>



Where to start?



VMware Event Broker Appliance

 VEBA

Home Documentation Functions Community FAQs Resources

Unlocking the Hidden Potential of Events in the VMware SDDC

Use event-driven automation and take your vSphere Events to the next level! Easily trigger custom or pre-built actions to deliver powerful integrations within your datacenter but also across public cloud services. Integrations like Slack, Pager Duty, Service Now, etc. has never been easier before

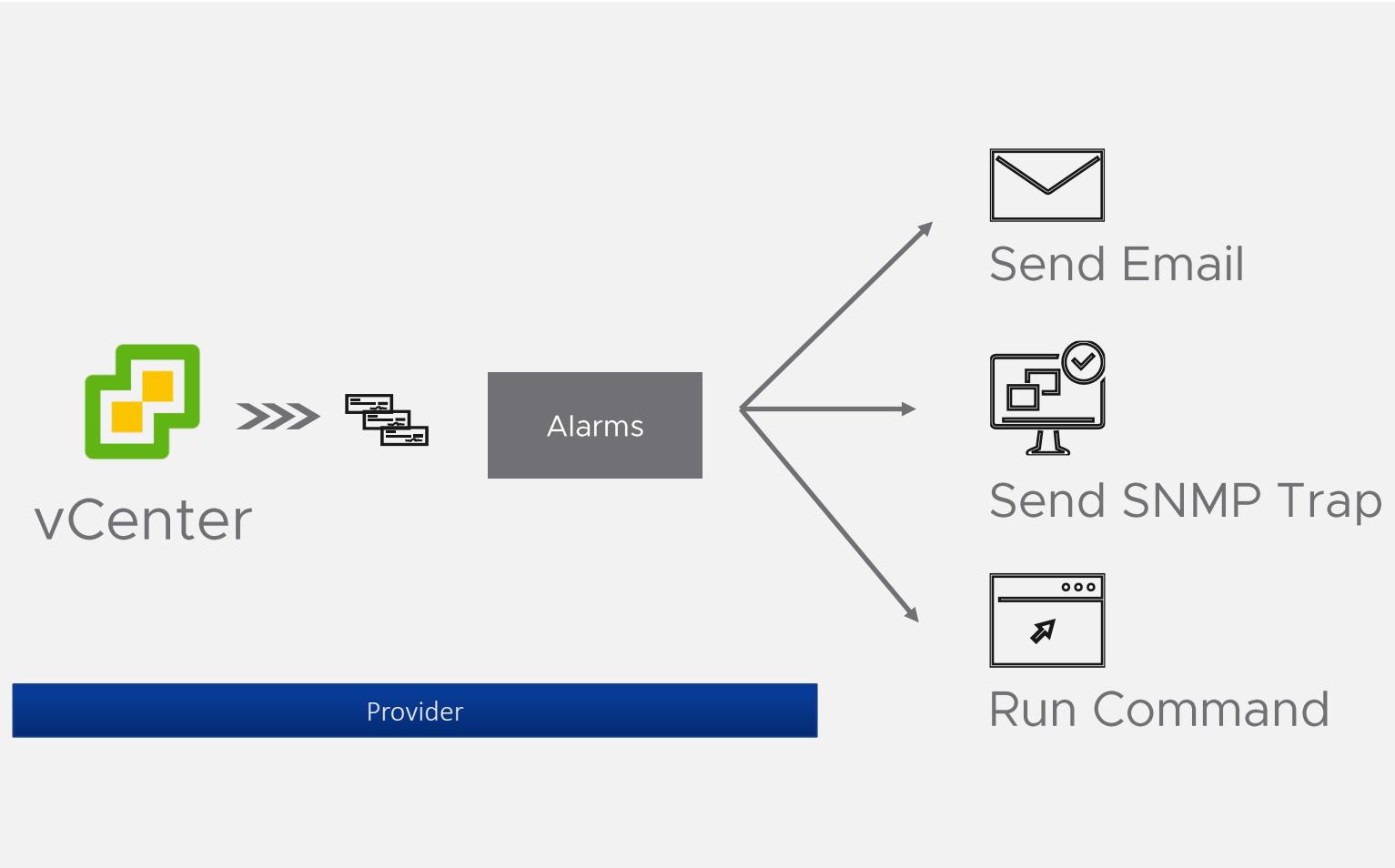
#VEBA brings a Day  Night of difference to your SDDC

[DOWNLOAD & INSTALL](#) [DEPLOY FUNCTIONS](#)



vCenter Event Driven Automation

Trigger Actions based on vCenter Events



Alarms allows limited and structured workflow capability

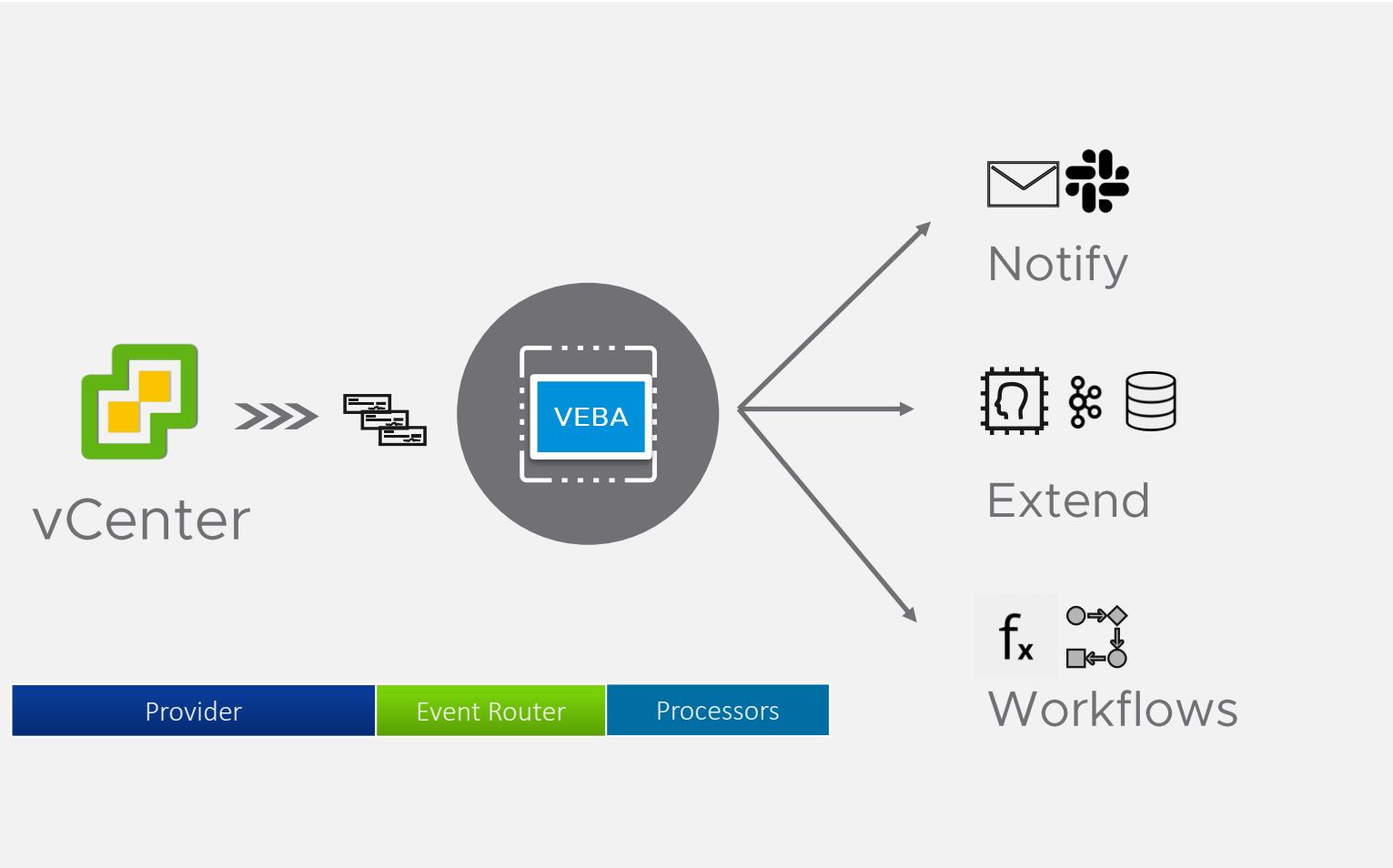
- Send Email
- Send SNMP traps
- Run Commands or Scripts

This limits innovation as this functionality needs

- vCenter access to setup Alarms
- Learning curve to implement and deploy code
- Redundant connectivity or platform code
- Added overhead on vCenter.

vCenter Event Driven Automation with VEBA

Trigger Actions based on vCenter Events



VMware Event Broker Appliance allows seamless extension of the vCenter platform

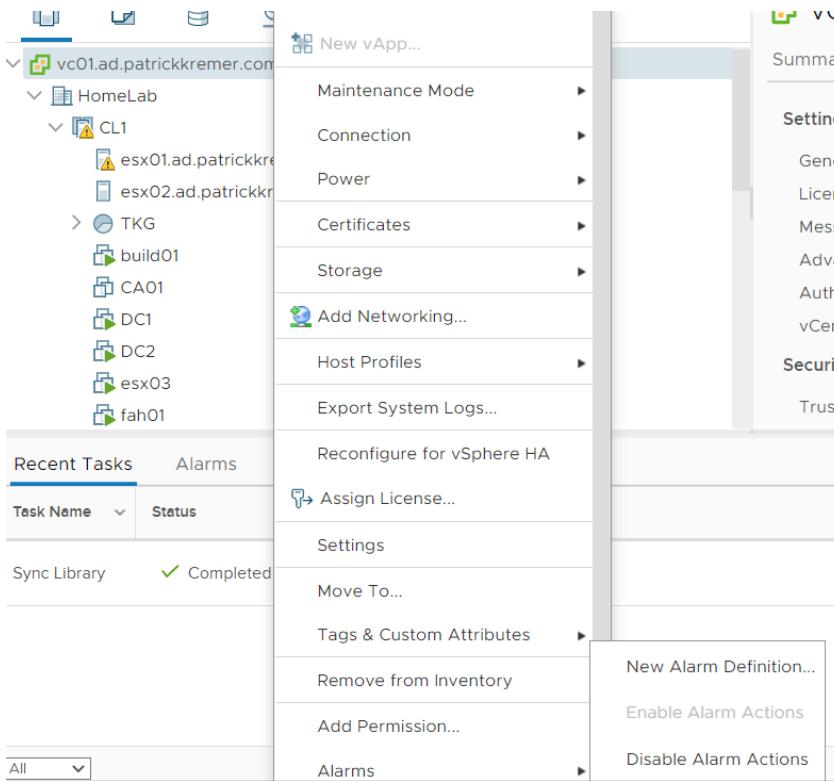
- Deployed as an appliance
- Enables innovation without requiring vCenter UI access
- Can enable fan-out messaging patterns

Function as a Service - Write and Execute code written in any language

Get started quickly with prebuilt functions maintained by the community.

Two examples of VEBA functions - PowerCLI

Maintenance Mode Alarms



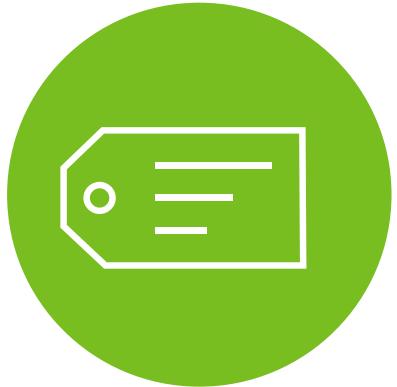
HA Event Notification

```
vsish -e set /reliability/crashMe/Panic 1
```

The screenshot shows an email message with a subject line of '** esx01 Failure - VMs Restarted **'. The message is addressed to 'noreply@vmware.com' and 'Patrick Kremer'. The body of the email contains a table with the following data:

ObjectName	Date	Description
build01	06/26/2020 14:12:23 -	
fah01	06/26/2020 14:12:29 - Version: 1.0.2 VMware FaH Team: 52737 B	
proxy01	06/26/2020 14:12:24 -	
tinylinux	06/26/2020 14:12:32 -	

Other VEBA functions



Tag a VM based
on a vCenter
event



Send a Slack
message when a
VM is
reconfigured



Notify Pagerduty
when a disk alarm
is tripped

November 2019 - VEBA can do that

Deploy OVF Template

✓ 1 Select an OVF template

2 Select a name and folder

3 Select a compute resource

4 Review details

5 Select storage

6 Ready to complete

Select a name and folder

Specify a unique name and target location

Virtual machine name: veba01

Select a location for the virtual machine.

vc01.ad.patrickkremer.com

HomeLab

VEBA Function Documentation

Do what with the what?

vSphere Datastore Usage Email Notification

Description

This function demonstrates using PowerShell to send an email notification when warning/error threshold is reach for Datastore Usage Alarm in vSphere

Consume Function Instruction

Step 1 - Clone repo

```
git clone https://github.com/vmware-samples/vcenter-event-broker-appliance  
cd vcenter-event-broker-appliance/examples/powercli/datastore-usage-email  
git checkout master
```

Step 2 - Update `stack.yml` and `vc-datastore-config.json` with your environment information

Stack.yml

Oh, and you need Docker too

```
! stack.yml X
examples > powercli > datastore-usage-email > ! stack.yml
      William Lam, 5 months ago | 1 author (William Lam)
1 provider:
2   name: openfaas
3   gateway: https://veba.primp-industries.com
4 functions:
5   powershell-datastore-usage:
6     lang: powercli
7     handler: ./handler
8     image: vmware/veba-powercli-datastore-notification:latest
9     environment:
10    write_debug: true
11    read_debug: true
12    function_debug: false
13    secrets:
14      - vc-datastore-config
15    annotations:
16      topic: AlarmStatusChangedEvent
17
```

Appliance Processes

```
root      5865  0.0  0.1 109004  8524 ?          S1    Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1.linux/moby/8d7b8ffdf3c9e7d5cf35bc033e3
root      5890  0.0  0.1 109004  8524 ?          S1    Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1.linux/moby/9627bde0be3a40678758bd78f0b
root      5896  0.0  0.1 109004  8580 ?          S1    Jul07  0:08 containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1.linux/moby/a31268d3d8314f594a220201169
root      5909  0.0  0.1 109004  8580 ?          S1    Jul07  0:08 containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1.linux/moby/eeec9af97b4823d25b269113e8c
root      5923  0.0  0.0 1024     4 ?          Ss    Jul07  0:00 /pause
root      5941  0.0  0.0 1024     4 ?          Ss    Jul07  0:00 /pause
root      5968  0.0  0.0 1024     4 ?          Ss    Jul07  0:00 /pause
root      5982  0.0  0.0 1024     4 ?          Ss    Jul07  0:00 /pause
root      6189  0.0  0.1 109004  10580 ?         S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/b50fc08d06ea774fc38d5df47a5
root      6206  0.5  1.2 216496  101572 ?         Ssl  Jul11  19:52 kube-controller-manager --authentication-kubeconfig=/etc/kubernetes/controller-manager.conf --authorization-kubeconfig=/etc/
root      6300  0.0  0.1 109004  8588 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/ae37aa9d28e21a38f3beec37118
root      6311  0.0  0.1 107596  8532 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/3320cdb04bc2f9b3617795f89c
root      6328  0.0  0.4 137624  37944 ?         Ssl  Jul11  1:29 contour serve --incluster --xds-address=0.0.0.0 --xds-port=8001 --envoy-service-http-port=80 --envoy-service-https-port=443
root      6340  0.0  0.5 141516  41956 ?         Ssl  Jul11  2:16 kube-scheduler --bind-address=127.0.0.1 --kubeconfig=/etc/kubernetes/scheduler.conf --leader-elect=true
root      6391  0.0  0.0 109004  8124 ?          S1   Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/892eb859f87fdfb58cc1290f40c
nobody    6429  0.0  0.0 106724  7900 ?          Ssl  Jul07  0:00 /usr/sbin/tinywww
root      6499  0.0  0.1 107596  8580 ?          S1   Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/d61aacca1669ecd1958951b422e
root      6531  0.0  0.0 12368   5024 ?         Ssl  Jul07  2:26 fwatchdog
root      6580  0.0  0.1 109004  8524 ?          S1   Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/9d0afcb6bbf8b83b1648fc660e5
root      6599  0.0  0.0 1024     4 ?          Ss    Jul07  0:00 /pause
root      7138  0.0  0.1 107596  8464 ?          S1   Jul07  0:09 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/736981b33d9171423d28bc67de3
100       7155  0.0  0.0 109252  8092 ?         Ssl  Jul07  0:22 ./app
root      7535  0.0  0.1 109004  8464 ?          S1   Jul07  0:18 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/b689c13c0673d3dba80d12d627d
10001    7553  0.0  0.2 110904  18616 ?         Ssl  Jul07  6:37 ./gateway
root      8024  0.0  0.1 107596  8524 ?          S1   Jul07  0:10 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/202d100329a095b7dd5826810f0
100       8043  0.0  0.1 111784  10288 ?         Ssl  Jul07  0:10 /home/app/faas-idler -dry-run=true
root      19990 0.0  0.0     0     0 ?           I    22:55  0:00 [kworker/0:1-ata_sff]
root      22049 0.0  0.0     0     0 ?           I    23:00  0:00 [kworker/0:0-ata_sff]
root      23408 0.0  0.0     0     0 ?           I    05:52  0:00 [kworker/u4:0-events_unbound]
root      24023 0.0  0.0     0     0 ?           I    23:05  0:00 [kworker/1:0-cgroup_pidlist_destroy]
root      24024 0.0  0.0     0     0 ?           I    23:05  0:00 [kworker/1:1-events]
root      24155 0.0  0.0     0     0 ?           I    23:05  0:00 [kworker/0:2-events_power_efficient]
root      25015 0.3  0.0  11780   6484 ?          Ss  23:08  0:00 sshd: root@pts/0
root      25024 0.3  0.0    7900   4740 pts/0        Ss  23:08  0:00 -bash
root      25099 0.0  0.0    6072   2280 pts/0        R+  23:08  0:00 ps aux
root      25350 0.0  0.1 109004  8472 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/9d7eac7e7ec43f3f725941dc52
root      25367 1.3  3.5 475412  286392 ?         Ssl  Jul11  49:04 kube-apiserver --advertise-address=192.168.203.39 --allow-privileged=true --authorization-mode=Node,RBAC --client-ca-file=/etc/kubernetes/cert/ca.pem
root      25404 0.0  0.1 107596  8472 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/bd5be0ea5fcc5f4fc46082f00fc
root      25422 0.1  0.4 142288  35784 ?         Ssl  Jul11  5:04 /coredns -conf /etc/coredns/Corefile
root      25429 0.0  0.1 107596  8472 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/37e587b1709661c193c02f2967d
root      25453 0.1  0.4 142288  35520 ?         Ssl  Jul11  4:37 /coredns -conf /etc/coredns/Corefile
root      25727 0.0  0.1 107596  8588 ?          S1   Jul11  0:05 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/8a6e4619ac89e5ea9b913582b48
61000    25745 0.0  0.2 809624  17516 ?         Ssl  Jul11  1:12 ./vmware-event-router -config /etc/vmware-event-router/event-router-config.json -verbose
root      25836 0.0  0.1 109004  12864 ?          S1   Jul11  0:03 containerd-shim -namespace moby -workdir /var/lib/containerd.io.containerd.runtime.v1.linux/moby/c840d8364ed64cc251ed285983d
root      25860 0.0  0.4 137624  38548 ?         Ssl  Jul11  2:35 contour serve --incluster --xds-address=0.0.0.0 --xds-port=8001 --envoy-service-http-port=80 --envoy-service-https-port=443
root@veba02 ~ % ^C
```

VEBA is
currently a
developer's
tool.

Me, November 26, 2019

VEBA is actually targeted at VI Admins

William Lam, November 27, 2019



William Lam is a Senior Staff Solution Architect working in the VMware Cloud team within the Cloud Services Business Unit (CSBU) at VMware. He focuses on Automation, Integration and Operation of the VMware Software Defined Datacenter (SDDC).

Timelines



Monday	Tuesday	Wednesday	Thursday	Friday
2	3	4	5	6
Re:invent 2019				
+6	+4	+5	+7	+9

December 5th, 2019 ▾

 **Patrick Kremer** 10:58 AM
This place is nuts. I'm in the bakery right next door for the moment

 **William Lam** 11:00 AM
walking over in a second

 1 

December 2019

Deploying VEBA

<http://www.patrickkremer.com/veba>

Edit

VMware Event Broker Appliance – Part I – Deployment

Introduction

I became aware of the [VMware Event Broker Appliance Fling \(VEBA\)](#) in December, 2019. The VEBA fling is open source code released by VMware which allows customers to easily create event-driven automation based on vCenter Server Events. You can think of it as a way to run scripts based on alarm events – but you're not limited to only the alarm events exposed in the vCenter GUI. Instead, you have the ability to respond to ANY event in vCenter.

What is Github?



GitHub is how people build software

We're supporting a community where more than 50 million* people learn, share, and work together to build software.

October 2007 First commit	San Francisco Headquarters	100 million* Repositories hosted
------------------------------	-------------------------------	-------------------------------------

What is a repository (repo)

[vmc-onboarding](#)

Forked from bohleadam/vmc-onboarding

An example documentation site using the Docsy Hugo theme

 HTML  242  Apache License 2.0 Updated 3 days ago

[vcenter-event-broker-appliance](#)

Forked from vmware-samples/vcenter-event-broker-appliance

The vCenter Event Broker Appliance enables customers to easily create event-driven automation based on vCenter Server Events

 Go  28  Other Updated 28 days ago

Impostor syndrome



Home Documentation Functions Community FAQs Resources

VMware Event Broker Appliance is released as open source software and provides community support through our GitHub project page. If you encounter an issue or have a question, feel free to reach out on the [GitHub issues page for VMware Event Broker Appliance](#).

Meet and get to know Team #VEBA.



Michael Gasch



William Lam



Frankie Gold



Partheeban Kandasamy (PK)



Patrick Kremer



Robert Guske



Vladimir Velikov



Martin Dekov

Here's them

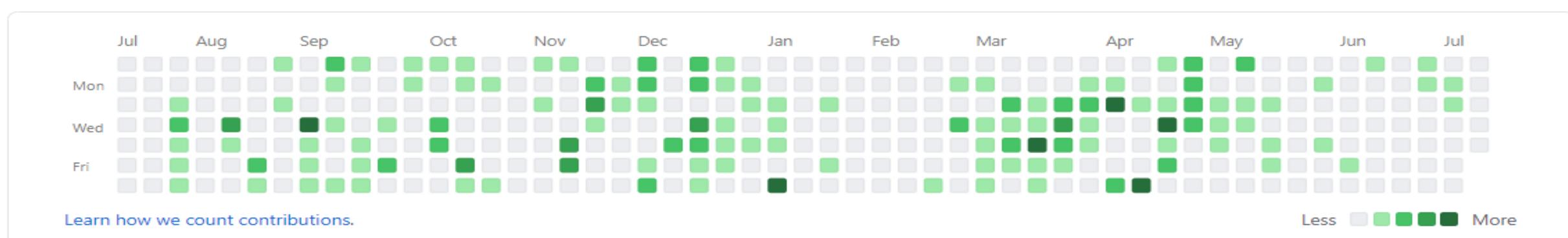
374 contributions in the last year



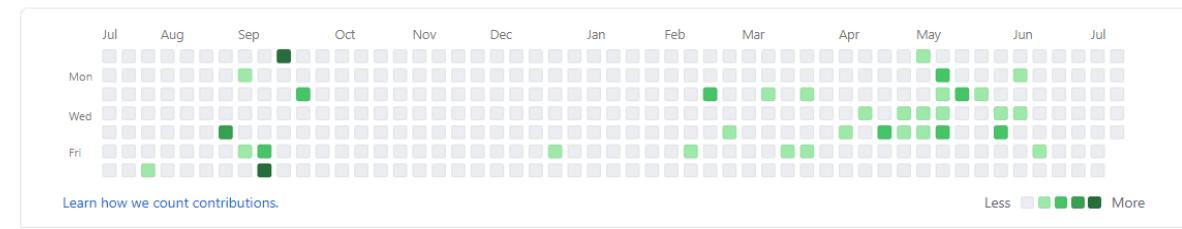
258 contributions in the last year



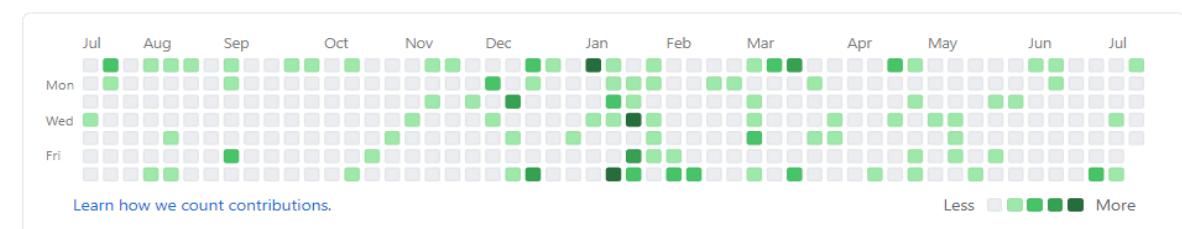
413 contributions in the last year



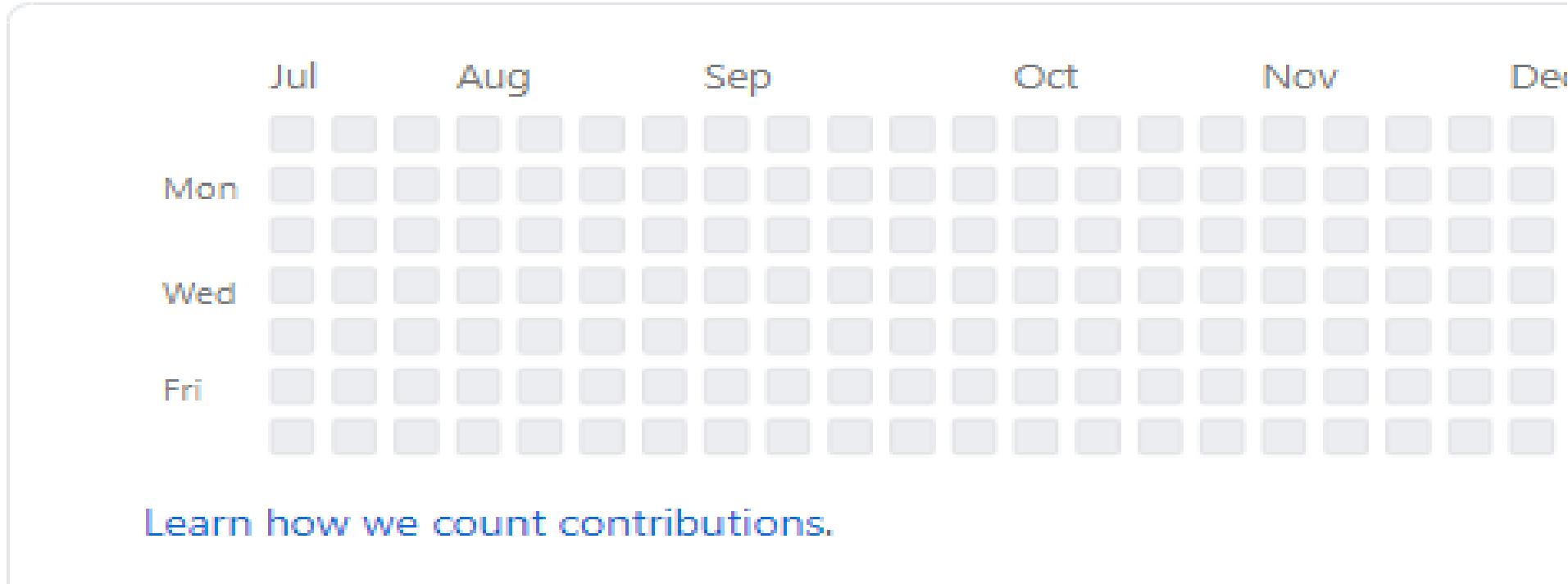
88 contributions in the last year



242 contributions in the last year



Here's me



Help from the community



Home Documentation Functions Community FAQs Resources

VMware Event Broker Appliance is released as open source software and provides community support through our GitHub project page. If you encounter an issue or have a question, feel free to reach out on the [GitHub issues page for VMware Event Broker Appliance](#).

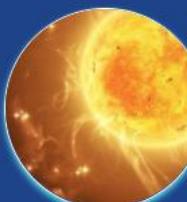
Meet and get to know Team #VEBA.



Michael Gasch



William Lam



Frankie Gold



Partheeban Kandasamy (PK)



Patrick Kremer



Robert Guske



Vladimir Velikov



Martin Dekov

Using the right tools

Containers run code

Pods are a group of containers

Pods are organized into namespaces

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
kube-system	coredns-584795fc57-4slh7	1/1	Running	19	119d
kube-system	coredns-584795fc57-5md8p	1/1	Running	18	119d
kube-system	etcd-veba02	1/1	Running	6	119d
kube-system	kube-apiserver-veba02	1/1	Running	16	119d
kube-system	kube-controller-manager-veba02	1/1	Running	108	119d
kube-system	kube-proxy-tsw54	1/1	Running	6	119d
kube-system	kube-scheduler-veba02	1/1	Running	105	119d
kube-system	weave-net-fkk7d	2/2	Running	18	119d
openfaas-fn	powercli-entermaint-794f6c66bf-s9gzk	1/1	Running	5	102d
openfaas-fn	powercli-ha-restarted-vms-679dcccd74-x2c5p	1/1	Running	1	17d
openfaas-fn	powershell-datastore-usage-597c96b584-bcfs8	1/1	Running	5	83d
openfaas	alertmanager-58f8d787d9-fmmn7	1/1	Running	6	119d
openfaas	basic-auth-plugin-dd49cd66b-lmnj2	1/1	Running	6	119d
openfaas	faas-idler-59ff9778fd-q68k8	1/1	Running	14	119d
openfaas	gateway-74f6f9489b-br5gz	2/2	Running	20	119d
openfaas	nats-6dfbf45d77-bblgf	1/1	Running	6	119d
openfaas	prometheus-5f5494b54f-hxfjt	1/1	Running	6	119d
openfaas	queue-worker-59b67bf4-rpqf9	1/1	Running	14	119d
projectcontour	contour-5cddfc8f6-57hhp	1/1	Running	68	119d
projectcontour	contour-5cddfc8f6-r2mxj	1/1	Running	78	119d
projectcontour	contour-certgen-7r9dl	0/1	Completed	0	119d
projectcontour	envoy-htrwv	1/1	Running	6	119d
vmware	tinywww-7fcfc6fb94-tv98j	1/1	Running	6	119d
vmware	vmware-event-router-5dd9c8f858-n9pg6	1/1	Running	53	96d

Using the right tools

Pods have logs!

```
root@veba02 [ ~ ]# kubectl logs vmware-event-router-5dd9c8f858-pcq2v -n vmware
[VMware Event Router] 2020/03/03 17:15:09 connecting to vCenter https://vc01.ad.patrickkremer.com/sdk
[VMware Event Router] 2020/03/03 17:15:09 connecting to OpenFaaS gateway http://gateway.openfaas:8080 (async mode: false)
[VMware Event Router] 2020/03/03 17:15:09 exposing metrics server on 0.0.0.0:8080 (auth: basic_auth)
[Metrics Server] 2020/03/03 17:15:09 starting metrics server and listening on "http://0.0.0.0:8080/stats"
2020/03/03 17:15:09 Syncing topic map
of topics: []
[OpenFaaS] 2020/03/03 17:15:09 processing event [0] of type *types.UserLogoutSessionEvent from source https://vc01.
```

I fixed a typo in Github

Monday	Tuesday	Wednesday	Thursday	Friday
2	3	4	5	6
Re:invent 2019				
+6	+4	+5	+7	+9

Fixed typo 'read_debuge -> read_debug' #15

→ Merged embano1 merged 1 commit into [vmware-samples:master](#) from [unknown repository](#) on Dec 10, 2019

Conversation 1 Commits 1 Checks 0 Files changed 1

Changes from all commits ▾ File filter... ▾ Jump to... ▾ ⚙ ▾

2 examples/python/tagging/README.MD

```
.... @@ -73,7 +73,7 @@ functions:  
73 73     image: embano1/pytag-fn:0.2  
74 74     environment:  
75 75     write_debug: true  
76 -     read_debuge: true  
76 +     read_debug: true  
77 77     secrets:  
78 78         - vcconfig # leave as is unless you changed the name during the creation of the vCenter credentials secrets above  
79 79     annotations:  
....
```

I fixed documentation

Fixes to documentation and sample config files for openfaas #18

Merged embano1 merged 1 commit into [vmware-samples:master](#) from [unknown repository](#) on Dec 16, 2019

Conversation 4 Commits 1 Checks 0 Files changed 7

Changes from all commits ▾ File filter... ▾ Jump to... ▾ 0 / 7 files viewed ⓘ

examples/powercli/hostmaint-alarms/README.md

Line	Change	Text
23	-	@@ -23,7 +23,7 @@ faas-cli secret create vcconfig --from-file=vcconfig.json --tls-no-verify
24	-	2. Update the gateway in the stack.yml file with your vCenter Event Broker Appliance address and deploy the functions.
25	-	```yaml
26	-	provider:
26	+	- name: faas
26	+	+ name: openfaas
27	-	gateway: https://veba.yourdomain.com
28	-	...
29	-	```
29	-

examples/powercli/hostmaint-alarms/stack.yml

Line	Change	Text
1	-	...
1	-	...
1	-	@@ -1,6 +1,6 @@
1	-	version: 1.0

Expanding my knowledge over time

[Part I - VMware Event Broker Appliance - Deployment](#)

[Part Ia - AWS EventBridge Deployment](#)

[Part II - VMware Event Broker Appliance - Sample Code Prereqs](#)

[Part III - VMware Event Broker Appliance - Tags and Clones](#)

[Part IV - VMware Event Broker Appliance - Deploying the First Sample Function](#)

[Part V - VMware Event Broker Appliance - Contributing to the VEBA Project](#)

[Part VI - VMware Event Broker Appliance - Syncing Your Fork](#)

[Part VII - VMware Event Broker Appliance - Deploy the Sample Host Maintenance Function](#)

[Part VIII - VMware Event Broker Appliance - Basic Troubleshooting Techniques](#)

[Part IX - Deploying the Datastore Sample Email Script in VMC](#)

[Part X - Building the Appliance OVA from source code](#)

[Part XI - Changing options in the OVA installer](#)

[Part XII - Advanced Function Troubleshooting](#)

Installation

vSphere	4 settings
vCenter Server	IP Address or Hostname of vCenter Server <u>vc01.ad.patrickkremer.co</u>
vCenter Username	Username to login to vCenter Server <u>administrator@vsphere.</u>

This is what the VEBA looks like during first boot

```
Configuring Static IP Address ...
Configuring NTP ...
Configuring hostname ...
Restarting Network ...
Restarting Timesync ...
Configuring root password ...
Retrieving vSphere & OpenFaaS Variables ...
Starting Docker ...
Disabling/Stopping IP Tables ...
Setting up k8s ...
-
```

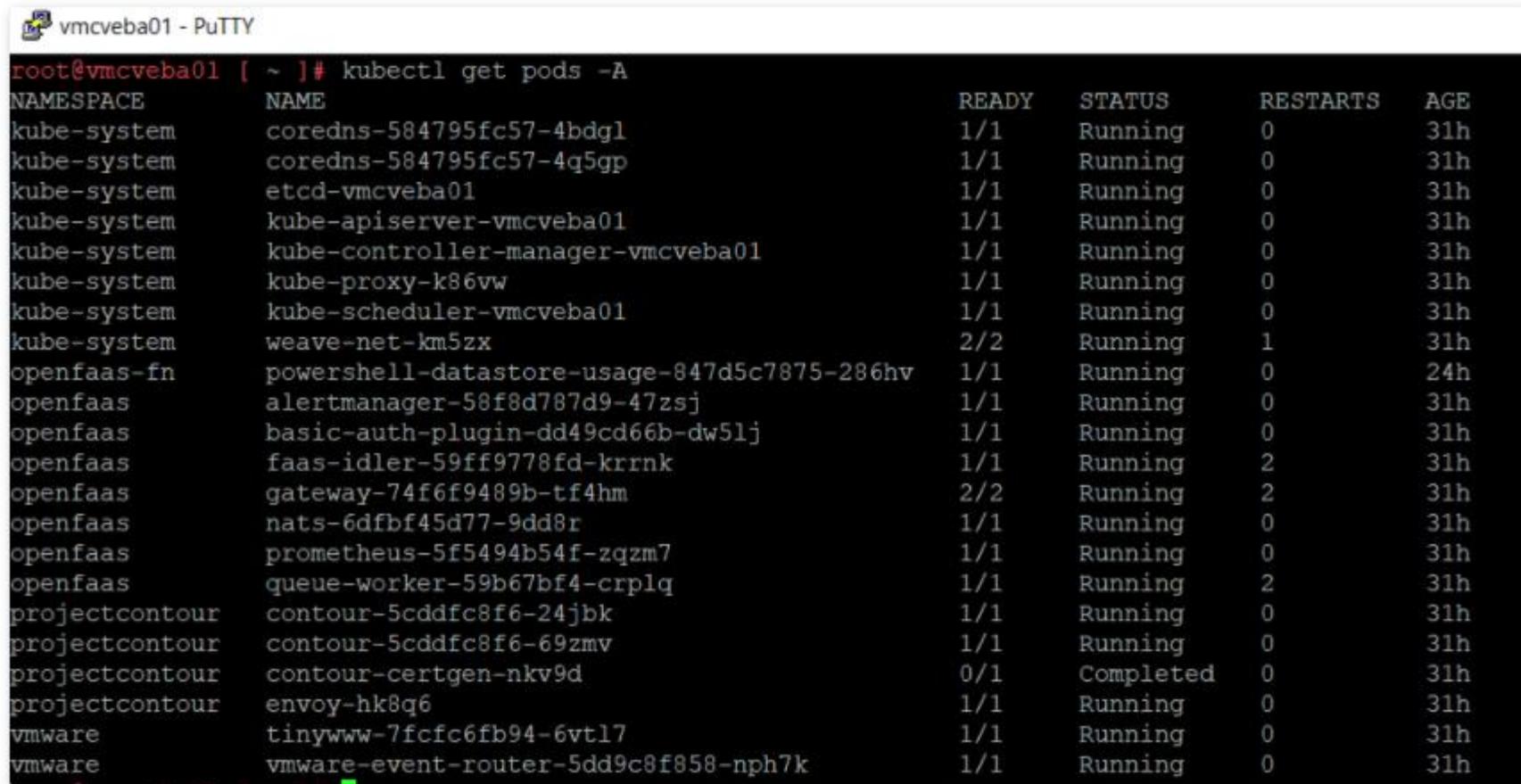
If you end up with this, [IP] in brackets instead of your hostname, something has failed. Did you use a subnet mask instead of CIDR? Did you put comma-separated DNS instead of space? Did you put in the incorrect gateway? If you enabled debugging at deploy time, you can look at /var/log/bootstrap-debug.log for detailed debug logs to help you pinpoint the error. If not, see what you can find in /var/log/bootstrap.log

Welcome to the vCenter Event Broker Appliance

Appliance Status: <https://veba02/status>
Install Logs: <https://veba02/bootstrap>
Appliance Statistics: <https://veba02/stats>
OpenFaaS UI: <https://veba02>

Troubleshooting

PROTIP: `kubectl get pods -A` gives you all the pods and their associated namespaces in one command.



A screenshot of a PuTTY terminal window titled "vmcveba01 - PuTTY". The command `kubectl get pods -A` is run, displaying a table of pod information across various namespaces. The columns are NAME, READY, STATUS, RESTARTS, and AGE. Most pods are in the "Running" status with 0 restarts, aged 31 hours. A few pods like "contour-certgen-nkv9d" are in "Completed" status.

NAME	READY	STATUS	RESTARTS	AGE
coredns-584795fc57-4bdgl	1/1	Running	0	31h
coredns-584795fc57-4q5gp	1/1	Running	0	31h
etcd-vmcveba01	1/1	Running	0	31h
kube-apiserver-vmcveba01	1/1	Running	0	31h
kube-controller-manager-vmcveba01	1/1	Running	0	31h
kube-proxy-k86vw	1/1	Running	0	31h
kube-scheduler-vmcveba01	1/1	Running	0	31h
weave-net-km5zx	2/2	Running	1	31h
powershell-datastore-usage-847d5c7875-286hv	1/1	Running	0	24h
alertmanager-58f8d787d9-47zsj	1/1	Running	0	31h
basic-auth-plugin-dd49cd66b-dw5lj	1/1	Running	0	31h
faas-idler-59ff9778fd-krrnk	1/1	Running	2	31h
gateway-74f6f9489b-tf4hm	2/2	Running	2	31h
nats-6dfbf45d77-9dd8r	1/1	Running	0	31h
prometheus-5f5494b54f-zqzm7	1/1	Running	0	31h
queue-worker-59b67bf4-crplq	1/1	Running	2	31h
contour-5cddfc8f6-24jbk	1/1	Running	0	31h
contour-5cddfc8f6-69zmv	1/1	Running	0	31h
contour-certgen-nkv9d	0/1	Completed	0	31h
envoy-hk8q6	1/1	Running	0	31h
tinywww-7fcfc6fb94-6vtl7	1/1	Running	0	31h
vmware-event-router-5dd9c8f858-nph7k	1/1	Running	0	31h

Git commands

```
pkremer@pkremer-w02 MINGW64 ~/OneDrive - VMware, Inc/VMware/git/vcenter-event-broker-appliance4 (master)
$ git diff
diff --git a/getting-started.md b/getting-started.md
index fbf9808..fe9f834 100644
--- a/getting-started.md
+++ b/getting-started.md
@@ -139,7 +139,7 @@ Lastly, define the vCenter event which will trigger this function. such function

```yaml
provider:
- name: faas
+ name: openfaas
 gateway: https://VEBA_FQDN_OR_IP # replace with your vCenter Event Broker Appliance environment
 functions:
 pytag-fn:
@@ -148,7 +148,7 @@ functions:
 image: embanol/pytag-fn:0.2
 environment:
 write_debug: true
- read_debug: true
+ read_debug: true
 secrets:
 - vcconfig # leave as is unless you ch
 annotations:
@@ -163,7 +163,7 @@ After you've performed th

```bash
faas-cli template pull # only required durin
-faas deploy -f stack.yml --tls-no-verify
+faas-cli deploy -f stack.yml --tls-no-verify
```

Everything looks good in the diff. Time to commit the code. I issue the command
git commit -a -s

-a stands for “all”, meaning we want to commit all changed files. I only changed one in this case, but if you changed multiple files, the -a switch is one way to commit all of them.

-s means that I’m signing the file with the user.name and user.email variables that we populated above.

When I issue the command, git pops open my text editor of choice. I need to write a comment documenting my changes. I write it, save it, then close the text editor.

Documentation, code, and more

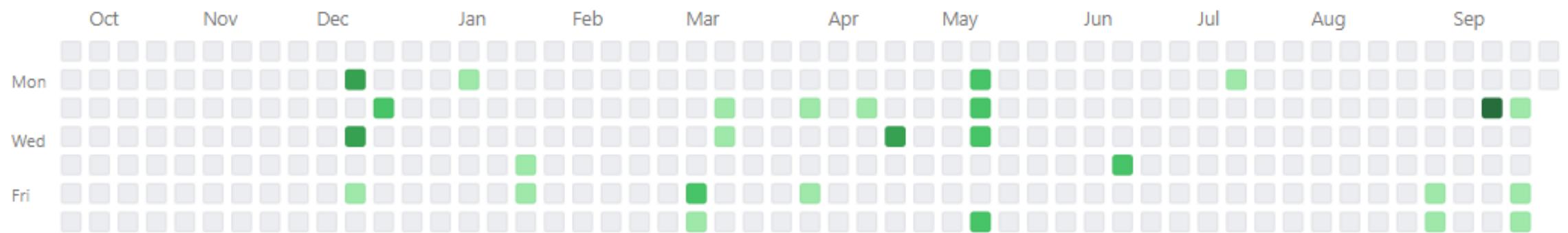
Total	Author
11	Aut
Updated Fling URL from vcenter to vmware ✓	#181 by kremerpatrick was merged on Jun 19 • Approved
Added enable SSH option to OVA install ✓	#147 by kremerpatrick was merged on May 16 • Approved
Added recommendation to add appliance IP to NO_PROXY	enhancement #142 by kremerpatrick was merged on May 14 • Approved
Updated docs to reflect new v0.4 path to event-router-config.json	documentation #137 by kremerpatrick was merged on May 14 • Approved
Added proxy support for deployment scripts	#135 by kremerpatrick was merged on May 12 • Approved
Support datastore custom attribute as To: address ✓	#111 by kremerpatrick was merged on May 15 • Approved
Add initial release of troubleshooting guide ✓	#93 by kremerpatrick was merged on Mar 31 • Approved
Added unauthenticated SMTP and green status emails	#72 by kremerpatrick was merged on Mar 7 • Changes requested v0.4 Release
Fixed provider name, read_debug, and faas-cli typo	#24 by kremerpatrick was merged on Dec 18, 2019

Contributing to open source

Github contributions

43 contributions in the last year

Contribution settings ▾

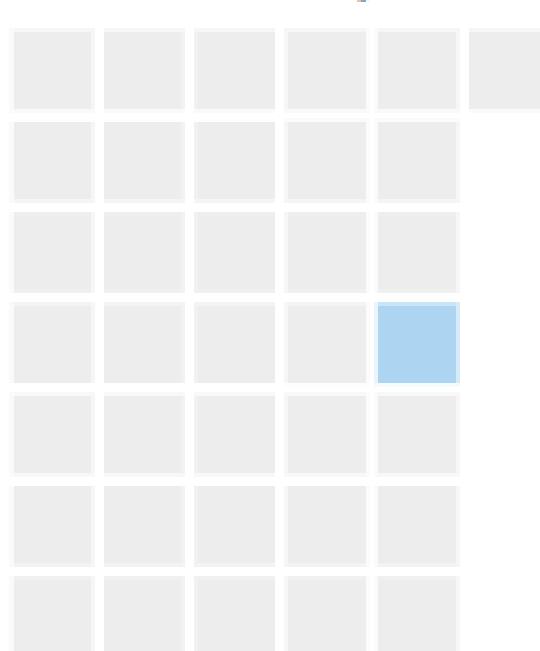


Learn how we count contributions.

Less More

Contributing to internal projects

Expanded visibility and opportunity with VMware



Sep

Activity

• **Patrick Kremer** @pkremer

Pushed to branch `master` at [https://github.com/pkremer/ansible-import-report](#)

`42d74270` · Added Confluence URL to readme

• **Patrick Kremer** @pkremer

Pushed to branch `master` at [https://github.com/pkremer/ansible-import-report](#)

`e742b1e5` · Added ~~Ansible support added in
prerelease~~ in the logs...

• **Patrick Kremer** @pkremer

Pushed to branch `master` at [https://github.com/pkremer/ansible-import-report](#)

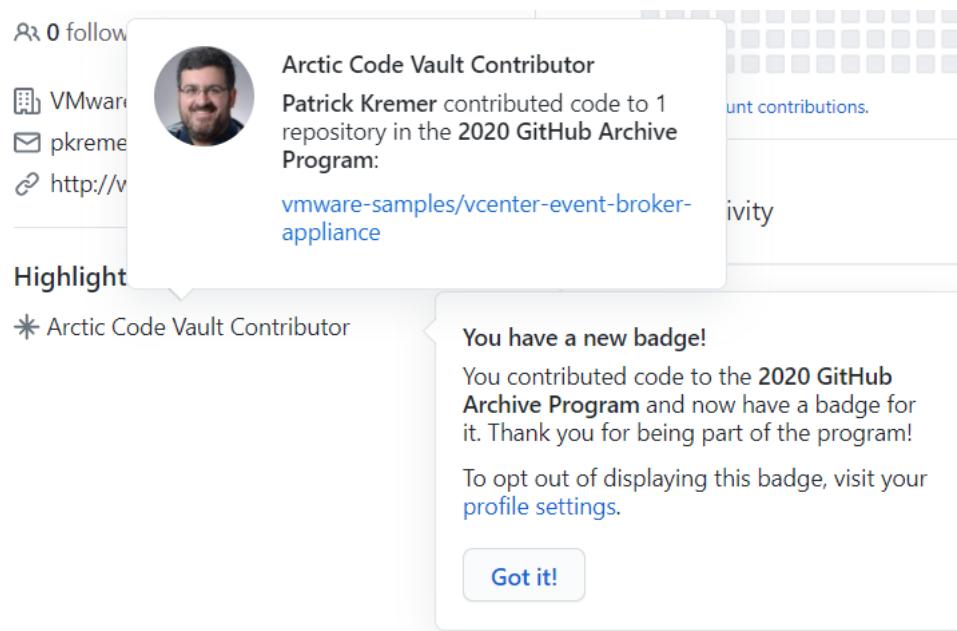
`30046f3e` · Added ~~Ansible support added in
prerelease~~

Contributing to open source

<https://archiveprogram.github.com/>

The GitHub Arctic Code Vault

The GitHub Arctic Code Vault is a data repository preserved in the Arctic World Archive (AWA), a very-long-term archival facility 250 meters deep in the permafrost of an Arctic mountain. The archive is located in a decommissioned coal mine in the Svalbard archipelago, closer to the North Pole than the Arctic Circle. GitHub will capture a snapshot of every active public repository on 02/02/2020 and preserve that data in the Arctic Code Vault.



xargs – Everything old is new again

```
⌄ 2 examples/powercli/hostmaint-alarms/template/powercli/template.yml
...
... @@ -1,3 +1,3 @@
1   1   language: powercli
2   - fprocess: xargs pwsh ./function/script.ps1
3   + fprocess: xargs -0 -I '{}' pwsh ./function/script.ps1 "'{}'"
3
```

How old is xargs? At least 40 years.

UNIX User's Manual

Release 3.0

T. A. Dolotta

S. B. Olsson

A. G. Petruccelli

Editors

June 1980

Not for use or disclosure outside the
Bell System except under written agreement.

XARGS(1)

NAME

xargs — construct argument list(s) and execute command

SYNOPSIS

xargs [flags] [command [initial-arguments]]

DESCRIPTION

Xargs combines the fixed *initial-arguments* with arguments read from standard input to execute the specified *command* one or more times. The number of arguments read for each *command* invocation and the manner in which they are combined are determined by the flags specified.

Command, which may be a shell file, is searched for, using one's **SPATH**. If *command* is omitted, **/bin/echo** is used.

Arguments read in from standard input are defined to be contiguous strings of characters delimited by one or more blanks, tabs, or new-lines; empty lines are always discarded. Blanks and tabs may be embedded as part of an argument if escaped or quoted: Characters enclosed in quotes (single or double) are taken literally, and the delimiting quotes are removed. Outside of quoted strings a backslash (\) will escape the next character.

Each argument list is constructed starting with the *initial-arguments*, followed by some number of arguments read from standard input (Exception: see **-i** flag). Flags **-i**, **-l**, and **-n** determine how arguments are selected for each command invocation. When none of these flags are coded, the *initial-arguments* are followed by arguments read continuously from stan-

XARGS(1)

Source: https://archive.org/details/bitsavers_attunixSysalRelease3Jun80_33886798

My checkbox

OS Credentials		2 settings
Root Password	Password to login in as root. Please use a secure password	
	Password	<hr/>
	Confirm Password	<hr/>
Enable SSH	Automatically start SSH daemon	
	<input type="checkbox"/>	<hr/>

My checkbox

```
9      - systemctl disable sshd
10     - systemctl stop sshd
9 + if [ "${ENABLE_SSH}" == "true" ]; then
10    + systemctl enable sshd
11    + systemctl start sshd
12  + else
13    + systemctl disable sshd
14    + systemctl stop sshd
15  + fi
...
21 21 ROOT_PASSWORD=$(vmtoolsd --cmd "info-get guestinfo.ovfEnv" | grep "guestinfo.root_password" | awk -F 'oe:value="" '{print $2}' | awk -F '"' '{print $1}')
22 + ENABLE_SSH=$(vmtoolsd --cmd "info-get guestinfo.ovfEnv" | grep "guestinfo.enable_ssh" | awk -F 'oe:value="" '{print $2}' | awk -F '"' '{print $1}' | tr '[upper:]'
22 23 VCENTER_SERVER=$(vmtoolsd --cmd "info-get guestinfo.ovfEnv" | grep "guestinfo.vcenter_server" | awk -F 'oe:value="" '{print $2}' | awk -F '"' '{print $1}')
...
62 62 </Property>
63 + <Property ovf:key="guestinfo.enable_ssh" ovf:type="boolean" ovf:userConfigurable="true" ovf:value="false">
64 +   <Label>Enable SSH</Label>
65 +   <Description>Automatically start SSH daemon</Description>
66 + </Property>
```

Create your own checkbox

VEBA Docs: <https://vmweventbroker.io/>

VEBA download: <https://vmware.com/go/veba>

My VEBA Blog: <http://patrickkremer.com/veba>

Public VEBA Slack:

<https://vmwarecode.slack.com/archives/CQLT9B5AA>





Thank You