**DRAFT\_v3: Intel News Article #2 (Final)**

**09/08/22**

**Words: 1599**

**Have You STIG’d Your Kubernetes Yet?**

*No? Why Not? Here’s How.*

*By: Andy Clemenko, Field Engineer, Rancher Government Solutions*

[andy.clemenko@rancherfederal.com](mailto:andy.clemenko@rancherfederal.com)

*Twitter:* [@clemenko](https://twitter.com/clemenko?ref_src=twsrc%255Egoogle%257Ctwcamp%255Eserp%257Ctwgr%255Eauthor)

Several weeks ago, I wrote about [Creating a Secure Kubernetes Deployment](https://intelligencecommunitynews.com/ic-insiders-creating-a-secure-kubernetes-deployment-five-ways-the-new-nsa-kubernetes-hardening-guide-can-help/), and five ways the NSA’s Kubernetes Hardening Guide could help. Now, I want to take the opportunity to go a little deeper and talk about Standard Technical Implementation Guides, or STIGs as they’re more commonly known.

Why? Because a STIG tells you *exactly* how to configure a system or application to a security standard. And in the Department of Defense (DoD) or Intelligence Community (IC), that’s critical. A STIG is written in conjunction with a vendor and with DISA, and once it’s written it is heavily reviewed by DISA *without the vendor’s input*. This is extremely important for validation purposes and ensures the STIG will be as effective as possible.

STIGs are important because they produce technical hardening guides that can be trusted. Today, many corporations are actually following the U.S. Government’s model and adopting STIGs as the hardening standard. It makes sense because STIGs effectively create a secure baseline for operating systems, frameworks, and applications, and applying a STIG dramatically decreases guesswork for IT staffs.

The DoD and IC have been using STIGs for years as the security baseline for all systems. They have also been encouraging vendors to work with DISA to write and improve STIGs for all network levels. In fact, we are even seeing STIGs being applied to lower Impact Levels (IL), such as IL5 and IL4.

In other words, STIGs have far-reaching value and impact.

***So, Where Can I Find STIGs?***

Generically speaking, you can find STIGs on [DISA’s site](https://public.cyber.mil/stigs/srg-stig-tools/). The STIG viewer can also be found on DISA’s [tools page](https://public.cyber.mil/stigs/srg-stig-tools/). For Kubernetes, there is a [generic k8s STIG](https://dl.dod.cyber.mil/wp-content/uploads/stigs/zip/U_Kubernetes_V1R6_STIG.zip).

But I want you to check out Rancher’s STIG. Rancher is the first company to have a multi-cluster Kubernetes manager STIG released by DISA. This is huge. It means we are protecting customers with a validated technical implementation guide. There is a nice article about it on[*Business Wire*](https://www.businesswire.com/news/home/20220425005143/en/DISA-Validates-Rancher-Government-Solutions%25E2%2580%2599-Security-Technical-Implementation-Guide-for-the-Rancher-Multi-cluster-Manager-2.6-for-Kubernetes)*.* (Good job, marketing team.)

You can also download the Rancher STIG itself.

***Ok. How Do I Apply the Rancher STIG?***

Thanks to the hard work of the Rancher engineers, there are only seven controls for the Rancher STIG. Please understand that this is for the Multi-Cluster Manager layer of the stack. We will talk about the Kubernetes layer later. Since there are only seven controls, let’s review them all.

**One:**

**V-252843 - Use an AUTH provider - Keycloak**

This control states that an external authentication method must be installed. A very popular one is called KeyCloak. [Rancher has great docs](https://rancher.com/docs/rancher/v2.6/en/admin-settings/authentication/keycloak-oidc/) that show how to configure OpenID Connect (OIDC).

**Two:**

**V-252844 - Audit Logging**

This control is to ensure Audit Logging is enabled. Fix Text: Ensure audit logging is enabled:

* Navigate to Triple Bar Symbol >> Explore Cluster >> local
* In the top Select ALL Namespaces from the drop down that currently says "Only User Namespaces"
* Click "deployments" under Workload menu item
* Select "rancher" in the Deployments section under the 'cattle-system' namespace.
* Click the three-dot config menu on the right
* Choose "Edit Config"
* Scroll down to the "Environment Variables" section
* Change the AUDIT\_LEVEL value to "2" or "3" and then click "Save"

If the variable does not exist:

* Click "Add Variable"
* Keep Default key/Value Pair as "Type"
* Add "AUDIT\_LEVEL" as Variable Name
* Input "2,3" for a value
* Click "Save"

Leveraging Helm is a great option for ensure everything is set. Simply add -set auditLog.level=2 --set auditLog.destination=hostPath to the Helm command or add the similar lines to the values.yaml file.

Here is an example of the command:

“helm upgrade -i rancher rancher-latest/rancher --create-namespace --namespace cattle-system --set hostname=rancher.$domain --set bootstrapPassword=bootStrapAllTheThings --set replicas=1 --set auditLog.level=2 --set auditLog.destination=hostPath”

**Three:**

**V-252845 - Role must be User**

This control is about adopting a tighter default user role for Rancher’s role-based access control (RBAC) – basically, to scope-in the default permissions. Fix Text: From the GUI, navigate to Triple Bar Symbol >> Users & Authentication. In the left navigation menu, click "Roles"

* Click "Standard User"
* At the top right, click the three dots, and then "Edit Config"
* Under "New User Default", select "No" and click "Save"
* Click "User-Base"
* At the top right, click the three dots, and then click "Edit Config"
* Under "New User Default", select "Yes", and then click "Save"

**Four:**

**V-252846 - Audit Record Storage**

This control is for installing and using logging for maintaining application logs. Fix Text: Enable log aggregation: Navigate to Triple Bar Symbol. For each cluster in "EXPLORE CLUSTER":

* Select "Cluster".
* Select "Cluster Tools" (bottom left)
* In the "Logging Block", select "Install"
* Select the newest version of logging in the dropdown
* Open the "Install into Project Dropdown"
* Select the Project (Note: Kubernetes STIG requires creating new project & namespace for deployments. Using Default or System is not best practice.)
* Click "Next"
* Review the options and click "Install"

**Five:**

**V-252847 - Never automatically remove or disable emergency accounts**

This control ensures that the local administrator emergency account is still active. Fix Text: Ensure local emergency admin account has not been removed and is the only Local account.

Navigate to the Triple Bar Symbol >> Users & Authentication. In the left navigation menu, click "Users." To Create a User:

* Click "Create"
* Complete the "Add User" form. Ensure Global Permissions are set to "Administrator"
* Click "Create"

To Delete a User:

* Select the user and click "Delete"

**Six:**

**V-252848 - Enforce organization-defined circumstances and/or usage conditions for organization-defined accounts.**

This control should be simple. Please ensure that you are an enterprise-signed certificate. Fix Text: Update the secrets to contain valid certificates. Put the correct and valid DOD certificate and key in files called "tls.crt" and "tls.key", respectively, and then run: “kubectl -n cattle-system create secret tls tls-rancher-ingress --cert=tls.crt --key=tls.key"

Upload the CA required for the certs by creating another file called "cacerts.pem" and running: “ kubectl -n cattle-system create secret generic tls-ca --from-file=cacerts.pem=./cacerts.pem”

The helm chart values need to be updated to include the check section: privateCA: true ingress: tls: ce: secret

Re-run helm upgrade with the new values for the certs to take effect.

**Seven:**

**V-252849 - Prohibit or restrict the use of protocols**

This control is for limiting the ports that are allowed for ingress to the Rancher UI/API.

Fix Text: Navigate to Triple Bar Symbol >> Explore Cluster >> local From the kubectl shell (>\_) execute the following:

kubectl patch -n cattle-system service rancher -p '{"spec":{"ports":[{"port":443,"targetPort":443}]}}'

# change the hostname to match your ingress URL.

export RANCHER\_HOSTNAME=rancher.rfed.io

kubectl -n cattle-system patch ingress rancher -p "{\"metadata\":{\"annotations\":{\"nginx.ingress.Kubernetes.io/backend-protocol\":\"HTTPS\"}},\"spec\":{\"rules\":[{\"host\":\"$RANCHER\_HOSTNAME\",\"http\":{\"paths\":[{\"backend\":{\"service\":{\"name\":\"rancher\",\"port\":{\"number\":443}}},\"pathType\":\"ImplementationSpecific\"}]}}]}}"

kubectl patch -n cattle-system service rancher --type=json -p '[{"op":"remove","path":"/spec/ports/0"}]'

***Really, Just Seven Steps?***

Really. Just seven steps.

***What About a STIG for RKE2?***

Soon, we will release the Rancher RKE2 STIG, which will also be simple to apply and with not a lot of knobs to turn. It’s presently under draft review by DISA and should be released by November. However, there are a few configurations users can tweak today to ensure RKE2 is hardened. Here’s how to do that:

First, it’s important to understand that RKE2 is basically broken down into servers and agents.

I’ll start with the server-side configurations. Begin by creating an Audit Policy. The following should be run on every server in the cluster.

```

cat <<EOT > /etc/rancher/rke2/audit-policy.yaml

apiVersion: audit.k8s.io/v1

kind: Policy

rules:

- level: RequestResponse

EOT

```

Next, look at the config file `/etc/rancher/rke2/config.yaml`. This file should be updated on every server node as well. Please note the Selinux setting. *(It goes without saying that enabling Selinux is important.)*

```

profile: cis-1.6

selinux: true

write-kubeconfig-mode: 0640

kube-controller-manager-arg:

- use-service-account-credentials=true

- tls-min-version=VersionTLS12

- tls-cipher-suites=TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_ECDSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384

kube-scheduler-arg:

- tls-min-version=VersionTLS12

- tls-cipher-suites=TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_ECDSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384

kube-apiserver-arg:

- tls-min-version=VersionTLS12

- tls-cipher-suites=TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,TLS\_ECDHE\_ECDSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305,TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384

- authorization-mode=RBAC,Node

- anonymous-auth=false

- audit-policy-file=/etc/rancher/rke2/audit-policy.yaml

- audit-log-mode=blocking-strict

kubelet-arg:

- protect-kernel-defaults=true

```

Once `/etc/rancher/rke2/config.yaml` is updated, simply use `systemctl restart rke2-server` to apply the changes.

So, that’s the server side of things. Now, let's look at the agent side.

The agent also uses `/etc/rancher/rke2/config.yaml`. Please notice that your `token-file` and `server` fields will be different. This is an example of what to set.

```

token-file: /etc/rancher/rke2/join\_token

server: https://$RKE\_SERVER:9345

write-kubeconfig-mode: 0640

profile: cis-1.6

kube-apiserver-arg:

- authorization-mode=RBAC,Node

kubelet-arg:

- protect-kernel-defaults=true

```

Once `/etc/rancher/rke2/config.yaml` is updated, simply use `systemctl restart rke2-agent` to apply the changes.

What Did We Learn?

Hopefully, I didn't bore you too much with all the technical talk! But it’s important to bridge the gap between the documentation, STIGs, and the knobs to turn. When it comes to STIG-ing your Kubernetes, please pass this article on to the technical folks, if that’s not you.

As a company, Rancher is dedicated to providing clients with secure Kubernetes and a secure multi-cloud manager. Several controls from Rancher and a handful of lines in a config file for RKE2 could not make this whole process any easier.

If we can help at all, or if you have questions about the code or instructions shared in this article, please don’t hesitate to reach out. And remember to keep a lookout for the DISA RKE2 STIG in November!

###

*The Author and Rancher Government Solutions assume no responsibility or liability for any errors or omissions, or for the results obtained from the use of this information. All information in this article is provided "as is.” Rancher Government Solutions is available to help with any implementation roadblocks.*