

# *Verbs*, not Nouns

## Writing Documentation Users Want to Read

Brian MacDonald  
KCDC 2023

brian.macdonald@upguard.com  
@bmac\_editor

# *Verbs*, not Nouns

## Writing Documentation Users Want to Read

Brian MacDonald  
KCDC 2023

brian.macdonald@upguard.com  
@bmac\_editor

# Two Types of Docs

- **Reference**
- **Tutorial**

# Reference docs

- Comprehensive
- Focus on the product
- Quick retrieval

# Reference docs



man pages  
API references  
code comments  
readme files

(V2) Get a list of questionnaire risks for one or more watched vendors or a specific questionnaire

Returns a list of currently open questionnaire risks for one or more of an account's watched vendors, including information on whether waivers exist for each risk.

## PARAMETERS

### Query Parameters ?

→ vendor_id	integer <int64> Restricts the questionnaire risks returned to a specific watched vendor by ID
→ primary_hostname	string Restricts the questionnaire risks returned to a specific watched vendor by vendor primary hostname
→ questionnaire_id	integer <int64> Restricts the questionnaire risks returned to a specific questionnaire by ID
→ page_token	string The token of the page to be returned. Will return the first page if left blank.
→ page_size	integer <int64> The number of risks to return per page. This Integer between 10 and 2000 defaults to 1000 if not supplied.
→ ignore_waived_risks	boolean Indicates that waived risks should not be returned in the risk set

GET /risks/vendors/questionnaires/v2

## RESPONSE SAMPLES

- 200 Questionnaire risks for specific watched vendors or questionnaire instances, including waiver details if appropriate
- 403 Account does not have access to requested entities (vendor, questionnaire)
- 422 An invalid parameter was supplied or a mandatory parameter was missing
- 500 An internal system error occurred

```
{
  "next_page_token": "4",
  - "risks": [
    + { ... }
  ],
  "total_results": 10
}
```

# Tutorial docs

- Not comprehensive
- Focus on the user
- Read beginning to end

# Tutorial docs

## Tutorials How-tos Walkthroughs Use cases

### CONTENTS

Prerequisites

Step 1 — Installing  
Apache

Step 2 — Adjusting  
the Firewall

Step 3 — Checking  
your Web Server

Step 4 — Managing  
the Apache Process

Step 5 — Setting Up  
Virtual Hosts  
(Recommended)

Step 6 — Getting  
Familiar with  
Important Apache  
Files and Directories

Conclusion

Once you're done setting this up, log in as your non-**root** user and proceed to the first step.

## Step 1 – Installing Apache

Apache is available within Ubuntu's default software repositories, making it possible to install it using conventional package management tools.

Begin by updating the local package index to reflect the latest upstream changes:

```
$ sudo apt update
```

Copy

Then, install the `apache2` package:

```
$ sudo apt install apache2
```

Copy

After confirming the installation, `apt` will install Apache and all required dependencies.

## Step 2 – Adjusting the Firewall

# Tutorials Tell a Story

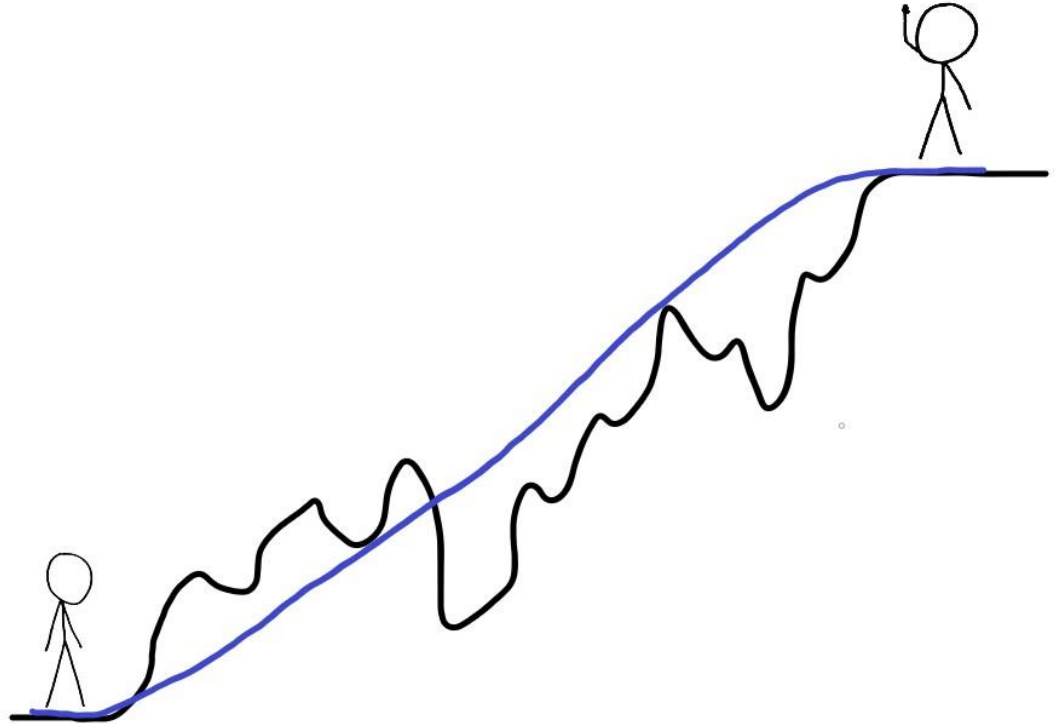
The user is the hero





# Tutorials Tell a Story

Tutorials smooth  
the user's path



# References are Nouns

**They tell you what something IS**

# When You Organize by Noun

## File menu



**New** - Opens a new document. If you use the keyboard combination indicated on the right a blank document opens immediately. Selecting the New menu item with your cursor gives the opportunity to open a large number of types of documents.

**Open** - Opens a previously saved document.

**Close** - Closes the active document but does not quit the application.

**Save** - Saves the active document with its current file name, location and format.

**Save As** - Saves by opening a window which gives the opportunity to change the file name, location or format.

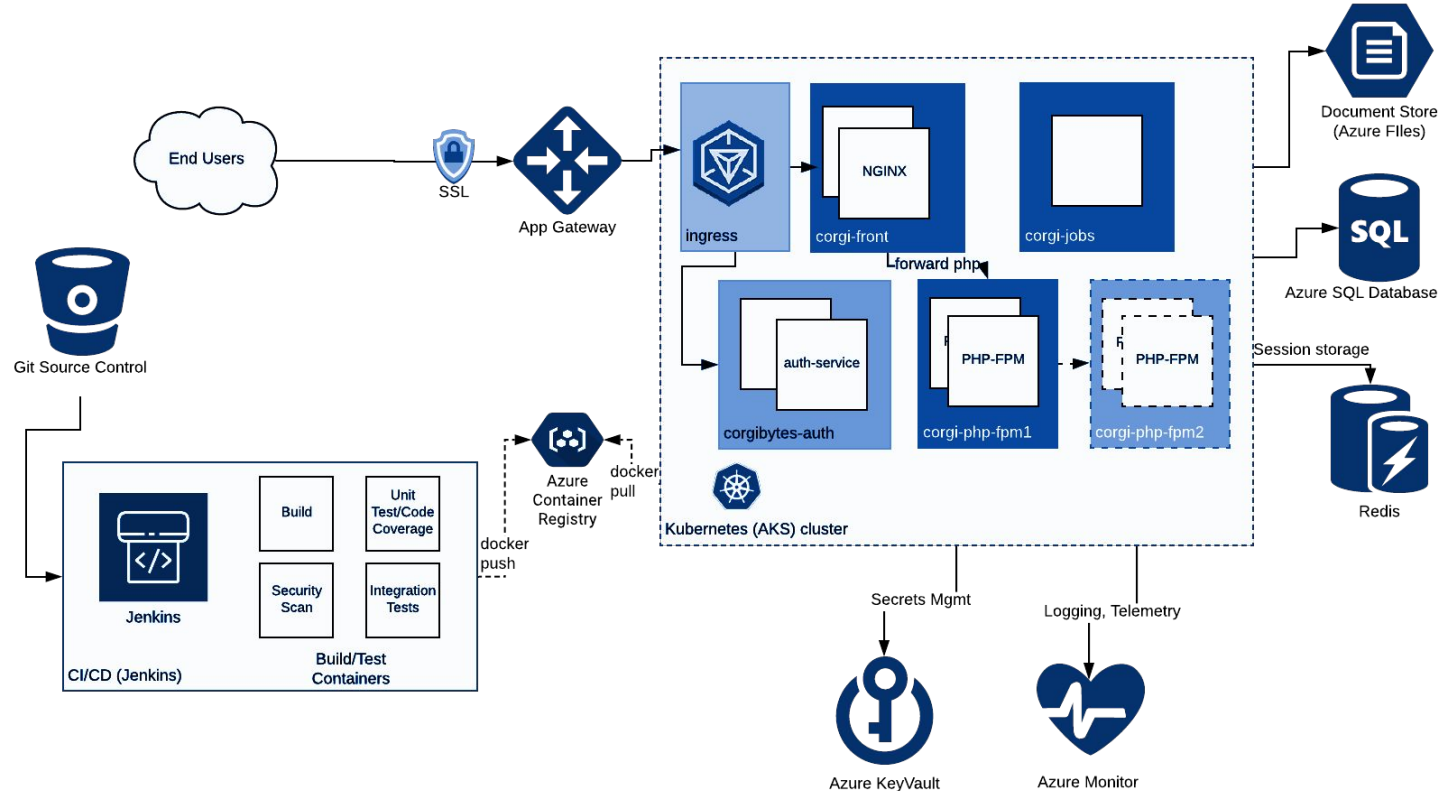
**Page Setup** - Sets margins, paper size, orientation and other layout options.

**Print Preview** - Shows how the file will look when you print it.

**Print** - Prints the active file, also gives the opportunity to change print options

**Exit** - Closes Microsoft Word.

# When You Organize by Noun



# Tutorials are Verbs

**They tell you how to  
DO something**

# When You Organize by Verb

## CONTENTS

Prerequisites

Step 1 — Installing Apache

Step 2 — Adjusting the Firewall

Step 3 — Checking your Web Server

Step 4 — Managing the Apache Process

Step 5 — Setting Up Virtual Hosts (Recommended)

Step 6 — Getting Familiar with Important Apache Files and Directories

Conclusion

Once you're done setting this up, log in as your non-**root** user and proceed to the first step.

## Step 1 – Installing Apache

Apache is available within Ubuntu's default software repositories, making it possible to install it using conventional package management tools.

Begin by updating the local package index to reflect the latest upstream changes:

```
$ sudo apt update
```

Copy

Then, install the `apache2` package:

```
$ sudo apt install apache2
```

Copy

After confirming the installation, `apt` will install Apache and all required dependencies.

## Step 2 – Adjusting the Firewall

# Not all Verbs are Equal

**Install**      **Create**      **Test**      **Automate**

**Configure**      **Monitor**      **Integrate**

**Deploy**      **Secure**      **Troubleshoot**

# Not all Verbs are Equal

**Know**

**Understand**

**Use**

**Learn**



# Not all Verbs are Equal

You'll learn how to add a server to your cluster.

You'll understand how to configure your firewall.

# Not all Verbs are Equal

You'll ~~learn how to~~ add a server to your cluster.  
You'll add a server to your cluster.

You'll ~~understand how to~~ configure your firewall.  
Configure your firewall.

# Grammar Pedantry Moment

Gerunds end in “-ing”

Install → Installing

Configure → Configuring

Troubleshoot → Troubleshooting

# How to Make Documentation

Don't just start writing...  
Brainstorm!

# Outline Your Doc First

- What do you want users to do?

# Outline Your Doc First

- What do you want users to do?
- Ask the users, if you can

# Outline Your Doc First

- What do you want users to do?
- Ask the users, if you can
- Write down everything

# Outline Your Doc First

- What do you want users to do?
- Ask the users, if you can
- Write down everything
- Define discrete steps



# Outline Your Doc First

- What do you want users to do?
- Ask the users, if you can
- Write down everything
- Define discrete steps
- Prioritize the use cases

# Live Demo Time

# Writing at the Step Level

# Writing at the Step Level

- Look for the verb

## How To View a Web Page

1. Enter the URL
2. Press Enter

# Writing at the Step Level

- Look for the verb
- Steps should be complete, and in order

## How To View a Web Page

1. **Start the browser**
2. **Click in the address bar**
3. **Type in the URL**
4. **Press Enter**

# Writing at the Step Level

- Look for the verb
- Steps should be in order
- Practice “just in time” learning

## How To View a Web Page

1. **Start the browser**
2. **Click in the address bar**
3. **Type in the web address**
  - a. A web address is also called a URL. It represents a location on the web to navigate to.
4. **Press Enter**

# Writing at the Step Level

- Look for the verb
- Steps should be in order
- Practice “just in time” learning
- Keep them short

## Step 1 – Installing Apache

Apache is available within Ubuntu’s default software repositories, making it possible to install it using conventional package management tools.

Begin by updating the local package index to reflect the latest upstream changes:

```
$ sudo apt update
```

[Copy](#)

Then, install the `apache2` package:

```
$ sudo apt install apache2
```

[Copy](#)

After confirming the installation, `apt` will install Apache and all required dependencies.

## Step 2 – Adjusting the Firewall

# Writing at the Step Level

- Look for the verb
- Steps should be in order
- Practice “just in time” learning
- Keep them short
- Use signposts



# Writing at the Step Level

- Look for the verb
- Steps should be in order
- Practice “just in time” learning
- Keep them short
- Use signposts
  - Introduction
  - Transitions
  - Conclusion

# Writing at the Step Level

The installation process will ask you about importing a GPG key. Confirm it so the installation can complete.

Now that you have Certbot installed, let's run it to get a certificate.

## Step 2 – Confirming Nginx's Configuration

Certbot needs to be able to find the correct `server` block in your Nginx configuration for it to be able to automatically configure SSL. Specifically, it does this by looking for a `server_name` directive that matches the domain you request a certificate for.

# Writing at the Paragraph Level

# Writing at the Paragraph Level

- Keep it concise

# Writing at the Paragraph Level

To add the Rocky Linux 9 EPEL repository, run the following command:

```
$ sudo dnf install epel-release
```

[Copy](#)

When asked to confirm the installation, type and enter `y`.

Now that you have access to the extra repository, install all of the required packages:

```
$ sudo dnf install certbot python3-certbot-nginx
```

[Copy](#)

This will install Certbot itself and the Nginx plugin for Certbot, which is needed to run the program.

The installation process will ask you about importing a GPG key. Confirm it so the installation can complete.

Now that you have Certbot installed, let's run it to get a certificate.

# Writing at the Paragraph Level

- Keep it concise
- Make it relevant

# Writing at the Paragraph Level

- Keep it concise
- Make it relevant
- Avoid subheadings

## Step 1. Installing Apache

```
sudo apt install apache2
```

## Step 2. Configuring Apache

- a. Adjusting the Firewall
- b. Checking the Server
- c. Managing the Processes
- d. Setting Up Virtual Hosts
- e. Configuring Log Files

Conclusion|

# Writing at the Paragraph Level

- Keep it concise
- Make it relevant
- Avoid subheadings

Step 1. Installing Apache

```
sudo apt install apache2
```

Step 2. Configuring Apache

Step 3. Adjusting the Firewall

Step 4. Checking the Server

Step 5. Managing the Processes

Step 6. Setting Up Virtual Hosts

Step 7. Configuring Log Files

Conclusion



# Writing at the Sentence Level

# Writing at the Sentence Level

- Look for the verb

The firewall settings are the next thing you need to change, so open the config file to do that.

# Writing at the Sentence Level

- Look for the verb

The firewall settings are the next thing you need to change, so **open the config file** to do that.

**Open the config file** to change the firewall settings.

# Writing at the Sentence Level

- Look for the verb
- Use active voice

The `mkdir` command is used to create a new directory.

# Writing at the Sentence Level

- Look for the verb
- Use active voice

The `mkdir` command **is used to** create a new directory.

**Create** a new directory with the `mkdir` command.

# Writing at the Sentence Level

- Look for the verb
- Use active voice
- Keep it short

Configure the firewall by opening the config file for editing, navigating to the allowed ports section, adding the ports that you want to open, and saving the file.

# Writing at the Sentence Level

- Look for the verb
- Use active voice
- Keep it short

Configure the firewall by **opening** the config file for editing, **navigating** to the allowed ports section, **adding** the ports that you want to open, and **saving** the file.

Configure the firewall by opening the config file for editing. **Navigate** to the allowed ports section. **Add** the ports that you want to open. Then **save** the file.

# Writing Tone



# Writing Tone

- Be confident casual

It is important to remember that security is critical to the functioning of a well-maintained system.

Therefore, plan your firewall configuration to allow access only on those ports essential to business needs.

# Writing Tone

- Be confident casual

~~It is important to remember that~~ security is critical to the functioning of a well-maintained system.

~~Therefore,~~ plan your firewall configuration to allow access only on those ports ~~essential to business needs~~.

Configure your firewall to open only the ports you need to.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”

Next, we'll create a dedicated user for the database instance.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”

Next, **we'll** create a dedicated user for the database instance.

Next, **you'll** create a dedicated user for the database instance.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”
- Use your audience’s vocabulary

You will create a modal that appears with a no-JavaScript approach using the `:target` pseudo class and the `opacity`, `pointer-events`, and `transition` properties.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”
- Use your audience’s vocabulary
- Avoid “easy” and “simple”

Configuring a new container in Kubernetes isn’t as hard as it sounds. You simply add the proper YAML to the kubectl file.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”
- Use your audience’s vocabulary
- Avoid “easy” and “simple”

Configuring a new container in Kubernetes ~~isn't as hard as it sounds~~. You ~~simply~~ add the proper YAML to the kubectl file.

To configure a new container in Kubernetes, add the proper YAML to the kubectl file.

# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”
- Use your audience’s vocabulary
- Avoid “easy” and “simple”
- No marketing copy

FooBar is a powerful new JavaScript framework that allows you to load complex UIs at lightning speed.



# Writing Tone

- Be confident casual
- Address the reader in second-person: “you”
- Use your audience’s vocabulary
- Avoid “easy” and “simple”
- No marketing copy

FooBar is a **powerful** new JavaScript framework that allows you to load complex UIs at **lightning** speed.

FooBar is a JavaScript framework that allows you to load complex UIs **more quickly**.

# Final Words



- **Look** for the verbs
- **Focus** on what the users want to do
- **Move** the verbs up front

Brian MacDonald  
KCDC 2023

brian.macdonald@upguard.com  
@bmac\_editor

Slides: <https://github.com/bmacdonald-editor/KCDC2023>

# WANNA WIN?



## SPEAKER FEEDBACK