Table of Contents

|  |  |
| --- | --- |
| Introduction | 3 |
| Success planning | 4 |
| Technical success | 9 |
| Adoption | 13 |
| Realized value | 15 |
|  |  |
|  |  |
|  |  |
|  |  |

#### 

#### Introduction

# Using STAR to guide your journey with Elastic

Welcome to the Elastic Customer Success STAR planning guide. STAR stands for the four steps you’ll take during this process:

* Success planning
* Technical success
* Adoption
* Realized value

Each section will go into more detail around the specific areas and assist you in mapping out what you need to do in order to get the most out of your investment in Elastic.

This is a planning guide and should be written in, scribbled on, and changed as you see fit. We have left plenty of space for you to write in your own specific needs. We recognize that every implementation is unique and has its own set of requirements. We recommend that as you fill out this guide you refer to the [list of features](https://www.elastic.co/subscriptions) available in your deployment as well as our [documentation](https://www.elastic.co/guide/index.html). This will help when it comes to mapping features to requirements.

## Notes:

Several of the tables in this document include sample responses in light gray and italics. Once you understand the flow, please feel free to remove the sample responses.

# Success planning

Before diving into the specific needs of your project from a technical and feature point of view, it is important to think critically about your direction of travel. Mapping your project back to the business needs of your organization will help you define value when you periodically review your project. Use this section to record those needs so they can be distributed and understood by your project stakeholders.

## Problem statement

|  |  |  |
| --- | --- | --- |
| **At a high level, what is the problem statement?** | | |
| *Automate the log collection, parsing, error identification, and issue assignment for the online banking service.* | | |

## 

## Current state

Defining your current state will give all the stakeholders an understanding of your starting point. You can also reflect on this section when it’s time to review your progress at your defined reflection point.

|  |  |  |
| --- | --- | --- |
| **How is the work done today?** | | |
| *Logs are sent via syslog to a central syslog server. They are stored for 90 days and purged. System admins manually search through logs using ssh, grep, etc.* | | |
| **What processes are in place today?** | | |
| *Employees ssh to the syslog server, manually find the right log, manually search for issues, cut and paste the log messages into a PagerDuty issue, look up the right team to assign, and manually fill in the hostnames, service name, etc. in PagerDuty.* | | |
| **What’s working?** | | **What’s not working?** |
| *We meet the regulatory requirements (90 day storage).* | | *The logs are used after an issue is reported, nothing is proactive.* |
|  | | *All of the manual work (looking up the proper team and service) is error prone.* |
|  | | *Sshing, grepping, etc. all takes time and some level of expertise — for example, figuring out what error code to search for.* |
| **How did you measure the effectiveness of your efforts?** | | |
| *We do not feel that we are efficient :( What we want to do is become proactive — for example, if there is a specific error message in a specific log, we want to be alerted and provide a runbook to operations that tells them how to deal with the error.* | | |

## 

## Stakeholders and goals

Writing out the groups involved in the project up front will help you define the resources and skills you need when you get to the **Technical success** and **Adoption** steps of this guide. Fill out the following table to define the teams, who they are, and how those teams are measured.

|  |  |  |  |
| --- | --- | --- | --- |
| **How is the organization structured? What are the various teams, and how do they divide responsibilities?** | | | |
| **Team** | | **Responsibilities** | **Stakeholders** |
| *System administrators* | | *Manage systems in their assigned location. Team is organized into two groups: Unix and Microsoft Windows* | * *Technical individual(s)* * *Management individual(s)* |
|  | |  |  |
|  | |  |  |

## 

|  |  |  |  |
| --- | --- | --- | --- |
| **What are the goals and measurements (how is “good” defined)?** | | | |
| **Team** | | **Goals** | **Measurements** |
| *System administrators* | | *Reduce downtime* | *Uptime of servers*  *Time to close high-severity issues* |
|  | |  |  |
|  | |  |  |

## 

## High-level plan

This section will help you define the goals for your project. It should be linked to the **Technical success** and **Adoption** parts of this document. Before completing this section, we recommend that you review first the **Technical success** and **Adoption** sections so you know what is expected there. This table should be easy to digest so that your project stakeholders can have a high-level understanding of your aims.

The following table is for all of the requirements. **Each row should have an ID, and that ID will carry through the rest of the sections** (Technical success, Adoption, and Realized value) so that you have an association between the requirement, how it is achieved, and how the value is measured and reported.

Double check the requirements that you enter here against the earlier tables. Group the requirements under the business goals as needed. In the example, the business goal “Meet GDPR requirements regarding customer privacy” is a grouping of two requirements. As you group the requirements you may see that the IDs are not in order in the table — that’s fine!

### 

### 

### 

### Business goals

### 

|  |  |  |  |
| --- | --- | --- | --- |
| **What are the business goals? (Copy from tables above)** | **Requirements: How can the solution support those goals?** | | **Requirement ID** |
| *Understand trends (server outages, outages by location, etc.)* | *Provide a report on count of high-severity issues by operating system group* | | *R1* |
| *Provide a report on count of L1 issues by location* | | *R11* |
| *Meet regulatory requirements for log storage* | *Store logs for 90 days* | | *R2* |
| *Integrate with DevOps processes* | *Send notifications via Slack* | | *R3* |
|  |  | |  |

## 

## Skills, training, and consulting

Understanding what skills the team has and where they can benefit from the resources available to them at Elastic is key to realizing the aims of your project. Once you have defined your requirements, return to this section to fill in the tables provided.

### Training

We have a multitude of training offerings designed to suit your needs and budget. The [Elastic Training page](https://www.elastic.co/training/) is a great place to learn about the free and paid options available. Use the table below to map out which trainings are necessary to the success of your project and the date by which they should be completed.

|  |  |  |  |
| --- | --- | --- | --- |
| **Team/Individual** | | **Link to training** | **Target completion** |
| *System Administrators* | | [*Observability Engineer*](https://www.elastic.co/training/elastic-observability-engineer) |  |
|  | |  |  |

## 

### Support

All Elastic customers have access to support in some form. Our Support team has put together a [guide for working with Elastic Support](https://www.elastic.co/support/welcome). Anyone on your team who will be working with support should familiarize themselves with the guide. If a group of individuals will be using support, please feel free to add this as a to do in the training table above.

### Consulting

Our Services team has a number of comprehensive consulting packages to help you accelerate your journey with Elastic. If you have purchased services, you can expect to engage with our Services team to schedule hours. If you are interested in our service offerings, please refer to the [Elastic Consulting page](https://www.elastic.co/consulting?ultron=EL-B-Stack-Trials-EMEA-UK-Exact&gambit=Elasticsearch-Consulting&blade=adwords-s&thor=elastic%20search%20consulting&gclid=EAIaIQobChMIjKOegeSO7AIV1untCh2MqALSEAAYAyAAEgJK0_D_BwE) for further details.

Prior to any services engagement, be sure to share your success plan with your consultant.

# 

# 

# Technical success

Most of the work required to successfully deploy your Elastic solution will be found in the documentation. Here, we will introduce a few core best practices. From there, please refer to the documentation specific to your use case.

* [Elastic Observability documentation](https://www.elastic.co/guide/en/observability/current/observability-introduction.html)
* [Elastic Security documentation](https://www.elastic.co/guide/en/security/current/es-overview.html)
* [Elastic Enterprise Search documentation](https://www.elastic.co/guide/en/enterprise-search/current/introduction.html)

## Use Elastic integrations

For many common sources, Elastic provides integrations that pick up, parse, and index data, enrich it with geo or other use case-specific information, and provide visualization capabilities. Take a look at the [integrations we provide](https://www.elastic.co/integrations), [integrations provided by the community](https://www.elastic.co/guide/en/beats/libbeat/current/community-beats.html), and [guidelines on creating your own integrations](https://www.elastic.co/guide/en/beats/devguide/current/index.html). Map any integration requirements (inbound or outbound) from your **Business goals** table to the existing lists, and identify any integrations that need to be created.

### Integration mapping

### 

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement name** | | **Integration details** | **Related requirement ID** |
| *Send notifications via Slack* | | *Alert using Slack webhook into the appropriate DevOps channel* | *R3* |
|  | |  |  |
|  | |  |  |

## 

## Use the [Elastic Common Schema](https://www.elastic.co/guide/en/ecs/current/ecs-getting-started.html) to support your workflow

Look back at the findings in the **Success planning** section and make sure that the data ingested into the Elastic Stack supports your workflow. For example, if your teams work based on geography, make sure that the proper field is populated with the information that supports your workflow. The examples in this table are based on an observability use case (IT operations logs, metrics, and application performance monitoring), but should be easily understood by people managing any other use case.

### Elastic Common Schema mapping

### 

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement name** | | **Elastic Common Schema mapping** | **Related requirement ID** |
| *Understand trends (server outages, outages by location, etc.)* | | *os.platform* | *R1* |
|  | |  |  |
|  | |  |  |

## 

## Use version control and secrets keystores for configuration

When you configure the Elastic Stack, you will need to specify sensitive settings or configurations such as passwords, endpoints, etc. Rather than relying on file system permissions to protect these values, you can use the Elastic product keystores to securely store values for use in configuration settings.

After adding a key and its secret value to the keystore, you can use the key in place of the secret value when you configure settings. In addition to increased security, moving environment-specific and secret configuration from configuration files to the keystores allows you to store configuration files in version control and reuse the same config files in dev, test, and production environments.

* [Elasticsearch keystore](https://www.elastic.co/guide/en/elasticsearch/reference/current/elasticsearch-keystore.html)
* [Kibana keystore](https://www.elastic.co/guide/en/kibana/current/secure-settings.html)
* [Filebeat keystore](https://www.elastic.co/guide/en/beats/filebeat/current/keystore.html)
* [Logstash keystore](https://www.elastic.co/guide/en/logstash/current/keystore.html)

## Detail stakeholder requirements

In the **Success planning** section you interviewed stakeholders and collected their requirements. Map those requirements to features. Include the field mappings from the Elastic Common Schema section above.

### Feature mapping

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement name** | | **Feature mapping** | **Related requirement ID** |
| *Provide a report on count of high-severity issues by operating system group* | | *Reporting, uses ECS field os.platform* | *R1* |
| *Assign work by operating system* | | *ECS field os.platform* | *R2* |
| *Store logs for 90 days* | | *Index lifecycle management: delete logs after 90 days.* | *R2* |
| *Send notifications via Slack* | | *Elastic Actions >* [*Slack Action*](https://www.elastic.co/guide/en/elasticsearch/reference/current/actions-slack.html) | *R3* |
|  | |  |  |
|  | |  |  |

## 

## Deployment phases

We suggest planning for phases from the start. Here’s why:

1. Finalizing the user experience should be done with real data in the system so that you can deploy user functionality (dashboards, user tools, etc.), get feedback from your users, and iterate.
2. Elastic adds new features often, and you should plan to review these on a regular basis. All releases are announced on the [Elastic blog](https://www.elastic.co/blog/category/releases).

Upgrading keeps you up to date with security, scalability, features, and supported versions.

## Phase schedule

|  |  |
| --- | --- |
|  | Review frequency |
| **Initial deployment** | *N/A* |
| **User experience updates** | *After initial deployment and then every three months* |
| **Features** | *Every three months* |
| **Upgrades** | *Every six months.* |

## 

### Deployment type and sizing

Your deployment type and sizing should mostly be already decided; however, there are some features that you may not be aware of that would be beneficial. As you get set up, it’s worth exploring things such as [setting up a cluster for high availability](https://www.elastic.co/guide/en/elasticsearch/reference/current/high-availability.html), [cross-cluster replication](https://www.elastic.co/guide/en/elasticsearch/reference/current/xpack-ccr.html), and [cross-cluster search](https://www.elastic.co/guide/en/elasticsearch/reference/current/modules-cross-cluster-search.html)

The deployment templates in [Elasticsearch Service](https://cloud.elastic.co/login) provide you with hot-warm architectures, high availability, compute optimized, and plenty of additional features. Elastic Cloud is a quick and easy way to get a deployment up and running, whether that be for testing, production, or monitoring.

Should you have any questions around sizing and architecture, we recommend getting in touch with Elastic Support, your Elastic account team, [Elastic Consulting](https://www.elastic.co/consulting/deployment), or one of our [partners](https://www.elastic.co/about/partners/).

# Adoption

Adoption comes in many forms and goes beyond feature adoption itself. In this section, we will look at adoption under three categories: administrator/user adoption, community adoption, and product adoption. Under each section we will provide guidance on how you can make strides to achieve all three.

## Administrator/user adoption

Enable your team of administrators and users with training and certification. Keep up to date with the latest release and align with your upgrading planning.

### Upgrades

Upgrades are an inevitable part of software adoption. Throughout your implementation, Elastic will keep you up to date with the latest features coming in future versions. As part of your implementation, you should build in time to review the latest version of Elastic, understand the features available, and upgrade when appropriate.

Build your upgrade strategy into your success plan under the **Stakeholders and goals** section above. However, there are a number of pieces to consider before you determine frequency and timing. As upgrading is such a fundamental task, we put together a blog post on [planning for success when upgrading the Elastic Stack](https://www.elastic.co/blog/upgrading-the-elastic-stack-planning-for-success), which you should link to in your plan for review prior to the upgrade process.

The plan in the blog post was written in collaboration with our Services team, who are the experts when it comes to upgrading. If you’d like to work with our Services team during your upgrade, [please contact Elastic Consulting](https://www.elastic.co/consulting/engage).

### Releases

Staying up to date with release can be hard. Setting aside some time to review the latest releases and features prior to upgrading will allow you to maximize the potential benefit you receive from upgrading. The easiest way to stay up to date on Elastic releases is by regularly checking the [release section of the Elastic blog](https://www.elastic.co/blog/category/releases).

In your success plan under the **Stakeholders and goals** section, record the frequency at which you plan to review features and identify any new features in later versions that would be of greatest benefit to your deployment. When the time comes to upgrade, make sure that your wider team and stakeholders are aware of the features you will be implementing and the benefit they will provide to your organization.

### Training

Making sure that your wider teams receive the education they need will also assist with adoption across the board. The **Skills, training, and consulting** section of this document covers the training that you may want your team to attend. For certain individuals in your organization, training may not be the best option — it may be better to guide them to [webinars](https://www.elastic.co/videos/) or [blog posts](https://www.elastic.co/blog/).

## 

## Community adoption

Understanding how to champion Elastic within your organization to broaden your network and win new use cases is a key to success. Build support at your organization by showing the success and value of current use cases as well as the capabilities of Elastic for future projects. There are two ways we do this at Elastic: by building a center of excellence (CoE) and through advocacy.

### Advocacy

Advocating for your project both internally and externally is a key tool to assist with adoption. One of the biggest advantages of working with Elastic is our extensive community. The community can support you in many ways. By far one of the most effective ways to engage is via either a user group or meetup. Our website provides an [overview of](https://www.elastic.co/events/) events that you can get involved in.

Profiling your use case with Elastic can also be a good way to get your story out there. The [User Stories](https://www.elastic.co/blog/category/user-stories) section of our blog is where we publish many of these profiles. If you wish to go this route, please do not hesitate to [get in touch](mailto:customersuccess@elastic.co).

## 

## Product adoption

Product adoption among your stakeholders and business users helps drive overall value. This comes in the form of feedback from users to help drive new features or helping users meet their own key performance indicators (KPIs). Product adoption is covered in this document under the **Success planning** and **Technical success** sections. Product adoption is fundamental to the success of your project, but is dependent on both community and administrator/user adoption.

# 

# Realized value

Realizing and communicating value isn't reserved for the period prior to the renewal of a contract. This is an ongoing process that must be consistently revisited during the development lifecycle.

Defining your KPIs at all levels in your success plan will give you a starter set of metrics to review at the intervals you deem appropriate. Defining the periods you review certain KPIs and pulling them to the front of your discussions is critical to the success of your project.

In this section, copy your business goals from the [high-level plan](#3y7ayfsq6iyj) into the table below and fill out each column. Set a review frequency for each of your goals and the associated KPIs. We recommend at least a quarterly progress review to determine where value is being created or lost. You can then determine whether you need to pivot or focus attention on a specific requirement.

|  |  |  |  |
| --- | --- | --- | --- |
| **Business goals** | **Review frequency** | **Target** | **Current state** |
| *Understand trends (server outages, outages by location, etc.)* | *After initial deployment and then every three months* | *Reduce downtime by 10%* | *Reduced by 5%* |
|  |  |  |  |
|  |  |  |  |

## 

If you are looking for inspiration on how other organizations have derived value from Elastic, read some of the [user stories](https://www.elastic.co/blog/category/user-stories) on our blog. Use the search bar to find use cases similar to yours that can help inspire your direction of travel.