**Synthetic Monitoring**

A screenshot of a computer

AI-generated content may be incorrect.

Synthetic monitoring is a proactive approach to website and application monitoring. Here's why it's crucial:

* **What is it?** Synthetic monitoring simulates user interactions with your website or application, running these simulations at regular intervals from various locations.
* **Why should you care?**
  1. Proactive Issue Detection: Identify problems before real users encounter them, improving overall user experience.
  2. 24/7 Monitoring: Constantly check your services, even during low-traffic periods.
  3. Consistent Testing Environment: Eliminate variables like user behavior or device types for reliable baseline metrics.
  4. Global Performance Insights: Test from multiple geographic locations to ensure consistent performance worldwide.
  5. Faster Time to Resolution: By simulating and recording user paths, you can quickly replicate and fix issues.
* **Real-world impact:** Businesses using synthetic monitoring can reduce downtime, improve user satisfaction, and ultimately protect their bottom line by ensuring their digital services are always performing optimally.

**Objective**

Create a basic synthetic monitor for Wikipedia's homepage using Elastic Synthetics.

**Private and Public Synthetics Locations**

When monitoring a public facing page, you want to ensure that it is reachable from many different locations around the world. When using Elastic Cloud, you can choose from a list of supported locations and run your checks from there. However, in this Lab we'll be running the checks from our own infrastrcuture. There are a few other differences between the two:

**Public Synthetic Monitors (Elastic Cloud)**

* Run from Elastic-managed locations around the world
* Ideal for monitoring public-facing websites and APIs
* Provide insights into global performance and availability
* No additional infrastructure required
* Limited to testing publicly accessible endpoints

**Private Synthetic Monitors**

* Run from your own infrastructure (on-premises or cloud)
* Suitable for monitoring internal applications, APIs, and services
* Allow testing of resources behind a firewall
* Offer greater control over test environment and network conditions

Creating your first synthetic monitor

We'll start simple and create a single page monitor first. A check of this kind allows you to monitor a single page only, but it's also really easy to set up.

1. To start with synthetic monitoring, we first need to create a new private location. Click on create Location.[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/a5cd4d61a58275d9df1c5753e0b3ce70/assets/CleanShot%202024-07-15%20at%2009.28.33%402x.png)
2. Give it the name Private and select the  Elastic Agent on ECK Policy.[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/5c87a64718d15c8edf6929568e68055a/assets/CleanShot%202024-07-15%20at%2009.28.53%402x.png)
3. Set the Website URL to

copy

https://www.wikipedia.org/

and then select the location private that you created earlier. Finally click on create monitor[A screenshot of a computer

AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/85fd4ef2df2716daed092f97371aab55/assets/CleanShot%202024-07-15%20at%2009.29.18%402x.png)

1. The synthetic monitor is now created. It will take a few seconds for the first run to finish. Click the refresh button at the top to refresh, until you see the test runs appear.[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/24cbd9069513492e7fa887071a6a8892/assets/CleanShot%202024-07-15%20at%2009.30.09%402x.png)[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/268729630182e58bf71202b8dc59bcd2/assets/CleanShot%202024-07-15%20at%2009.30.28%402x.png)
2. Click on the synthetic monitor.[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/42e858199c56f8091fbfd7f741077dd8/assets/CleanShot%202024-07-15%20at%2009.30.43%402x.png)
3. You should now see that it succeeded and its availability is at 100%[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/7b7f758eab61a67cc032d8cb3e12e05d/assets/CleanShot%202024-07-15%20at%2009.30.59%402x.png)
4. If you scroll down slightly you can also see a screenshot of the page which was captured as part the execution. This is useful for troubleshooting issues later on.[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/a7a2317a177a2cbe3de0e756765b2c01/assets/CleanShot%202024-07-15%20at%2009.31.11%402x.png)

Multi-step monitors are a powerful feature of synthetic monitoring. Here's why they're crucial:

* **What are they?** Multi-step monitors simulate complex user journeys by executing a series of steps or interactions in sequence, often including element checks along the way.
* **Why should you care?**
  1. End-to-End Testing: Verify entire user flows, not just individual pages.
  2. Critical Path Monitoring: Ensure key business processes (e.g., checkout, registration) are always functional.
  3. Detailed Diagnostics: Pinpoint exactly where in a process issues occur.
  4. Performance Benchmarking: Measure performance across different steps of user interaction.
  5. Realistic User Simulation: Mimic actual user behavior more closely than single-page checks.
* **Element checking within multi-step monitors:**
  1. Verifies that necessary elements are present and functional at each step.
  2. Ensures the journey can proceed as expected (e.g., a 'Next' button is clickable).
  3. Helps identify issues with dynamic content loading or state changes between steps.
* **Real-world impact:** Multi-step monitors can significantly improve the reliability of complex web applications. By ensuring that entire user journeys work as expected, businesses can reduce cart abandonment, improve conversion rates, and enhance overall user satisfaction.

Create a multi page synthetic monitor for Wikipedia's homepage using Elastic Synthetics.

Updating your synthetic monitor

Your next goal is to make the check more sophisticated. We would like to monitor multiple pages and check for the presence of an element on the page.

1. Hover over the previously created monitor, then click edit Monitor[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/a18776fb2978acb056b6088e22727910/assets/CleanShot%202024-07-15%20at%2010.22.38%402x.png)
2. Replace the contents of the Script editor with the following:[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/31c220961cfb5e0dc19148096ff8c0f3/assets/CleanShot%202024-07-15%20at%2009.35.16%402x.png)

copy

step('Go to https://www.wikipedia.org/', async () => {

page.setDefaultTimeout(10000)

await page.goto('https://www.wikipedia.org/');

});

step('Search Elasticsearch', async () => {

page.setDefaultTimeout(10000)

await page.getByLabel('Search Wikipedia').click();

await page.getByLabel('Search Wikipedia').fill('elasticsearch');

});

step('Open Elasticsearch, check github link', async () => {

page.setDefaultTimeout(10000)

await page.getByLabel('Search Wikipedia').press('Enter');

await page.getByRole('cell', { name: 'github.com/elastic/elasticsearch' });

});

This multi step monitor goes to wikipedia, searches for Elasticsearch, opens the result and then checks the final page for a cell with the content github.com/elastic/elasticsearch

1. Scroll down slightly and click the Run test button[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/3ab61d0b94ed0e7a158a315dd18a928e/assets/CleanShot%202024-07-15%20at%2009.35.20%402x.png)
2. Wait for the test run to finish[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/d9d350e2eb1176b6db50b652950f2238/assets/CleanShot%202024-07-15%20at%2009.35.27%402x.png)[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/99e4a6696ad81c2997b68da5b0c2b8cd/assets/CleanShot%202024-07-15%20at%2009.35.48%402x.png)then close the panel[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/ecefb09b4faeff9f3ec5196ad5f2ea30/assets/CleanShot%202024-07-15%20at%2009.36.13%402x.png)
3. Now that we know that the monitor works as expected, let's update it by clicking the Update monitor button[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/7eefd07766461869791e8a452e068663/assets/CleanShot%202024-07-15%20at%2009.36.21%402x.png)
4. And click on our monitor again[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/6eb76cb95118b31b083d07062edb07c2/assets/CleanShot%202024-07-25%20at%2019.34.12%402x.png)
5. Scrolling down a little, we can inspect our flow more closely. As you can tell, we're now taking three screenshots and capturing detailed data for each. Let's click on the last one[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/75c7c0a256f56b899e3b01156334cce2/assets/CleanShot%202024-07-15%20at%2009.36.51%402x.png)
6. In this view we can see the page load duration in lots of detail[A screenshot of a computer

   AI-generated content may be incorrect.](https://play.instruqt.com/assets/tracks/lrqnsbcyrshq/7c1d7e4129fee2be11f67c4557f1a282/assets/CleanShot%202024-07-15%20at%2009.37.02%402x.png)

Capturing screenshots during errors is a powerful diagnostic tool. Here's why it's valuable:

* **What is it?** Error screenshots automatically capture an image of the webpage when a monitored interaction fails or an error occurs.
* **Why should you care?**
  1. Visual Context: Provides immediate visual information about what the user sees during an error.
  2. Faster Debugging: Quickly identify UI-related issues that may not be apparent in logs or code.
  3. Cross-team Communication: Easily share visual evidence of issues with non-technical team members.
  4. Historical Record: Keep a visual log of how errors appeared over time, useful for tracking intermittent issues.
  5. User Experience Insight: Understand exactly what users encounter when things go wrong.
* **Real-world impact:** Error screenshots can significantly reduce time-to-resolution for issues, improve communication between teams, and help prioritize fixes based on visual severity, ultimately leading to better overall service quality.