

Advanced Scenarios and Solutions

This chapter addresses complex synthetic monitoring scenarios with step-by-step explanations. These include dynamic input handling, secure password management, popup interaction, conditional workflows, external integrations, proxy authentication, and monitor chaining.

1 Handling Dynamic Search Values in Synthetic Health Checks

Scenario:

You need to supply a dynamic value in a search box, which returns varying results each time. The challenge is monitoring this reliably using synthetic health checks.

Solution:

1. Use a **Clickpath Browser Monitor**.
2. Launch the **Clickpath Recorder**.
3. Simulate typing into the search field using a **parameterized or static value** (Dynatrace doesn't support true dynamic variables fetched at runtime yet).
4. Insert **validation checks** post-search to ensure:
 - Page loaded successfully
 - Search results section is rendered
 - Certain UI elements (e.g., "result count", "first item") appear
5. If dynamic content varies, use **smart selectors** or validation that accepts presence of partial content.
6. Optionally integrate with external scripts via **custom synthetic scripts** using Dynatrace CLI (for CI/CD automation) if truly dynamic values are needed.

Note: Full runtime dynamic value injection is limited. For advanced control, external scripting via CI/CD pipeline is preferred.

2 Password Vault in Dynatrace

Use Case:

Securely manage credentials used in synthetic monitoring, such as login forms or API tokens.

Solution:

1. Navigate to **Settings > Preferences > Credential Vault**.
2. Click **Add Credential**:
 - Choose type (Username/Password or Token)

- Enter secret values
 - Set appropriate **access scope** (e.g., Management Zone restrictions)
3. In your **Clickpath or HTTP monitor**, reference the vault credentials:
- For browser steps: use the credential selector in the input field.
 - For HTTP monitors: reference in the headers or body.

Credentials are securely injected at runtime and not exposed in UI or logs.

3 Handling JavaScript Popup Windows

Scenario:

Your application triggers popup modals or windows via JavaScript (e.g., confirmations, alerts).

Solution:

1. Use a **Clickpath Browser Monitor**.
 2. Record the user journey including popup interactions.
 3. Dynatrace supports handling:
 - **In-page modals**: simulate button clicks on modal
 - **Browser-native popups**: handled implicitly if part of the click flow
 4. Insert **validation rules** to check if popup opened and closed successfully.
 5. If popup uses dynamic selectors, use **smart wait** or content match for detection.
-

4 Iteration and Conditional Flows

Scenario:

You need to loop through HTML elements or take actions based on the state of a page (e.g., click next only if item is available).

Solution:

Dynatrace currently does not support **scripting logic (loops/conditionals)** directly inside Clickpath Monitors. However:

1. You can insert **multiple validation points** and **fallback steps**.
 2. Use **Smart Wait** and **Element Availability checks**.
 3. For full control, use Dynatrace **Custom Synthetic Monitoring via CI/CD integration** or external browser scripting tools.
-

5 External Code Execution & Integration with Vaults

Scenario:

Passwords or tokens are stored in an external vault (e.g., HashiCorp, AWS Secrets Manager), and must be injected dynamically.

Solution:

1. Use **external script execution** in your CI/CD pipeline:
 - Write a script that fetches secrets from the external vault.
 - Inject these into Dynatrace via API or environment variables.
2. Use Dynatrace's **Settings API or Configuration-as-Code CLI tool** to update credentials.
3. Schedule synthetic monitor updates via GitOps or Jenkins pipelines.

Dynatrace does not natively pull secrets from third-party vaults at runtime but supports integration via DevOps workflows.

6 Proxy Configuration for Online URL Monitoring

Scenario:

You need to monitor external URLs that require proxy access with authentication.

Solution:

1. Deploy an **ActiveGate** configured with a **custom proxy setup**.
 2. Edit the ActiveGate configuration file:
 - Add proxy host, port, and credentials.
 - Restart the ActiveGate.
 3. In **synthetic monitor settings**, ensure the execution location is set to the ActiveGate.
 4. Test connectivity and monitor execution to confirm proxy routing.
-

7 Chaining Monitors (e-Business Transaction Simulation)

Scenario:

You want the output from one monitor (e.g., session ID or result set) to feed into another.

Solution:

Dynatrace does not natively support monitor chaining **within the UI**, but it can be achieved via external orchestration:

1. Use a **CI/CD pipeline** or **external script**:

- Execute Monitor A via API.
 - Extract required output (e.g., response body or token).
 - Inject that into Monitor B's configuration using Dynatrace API.
2. Chain steps inside an **HTTP Monitor** using extracted variables.
- Step 1: Authenticate and store session token
 - Step 2: Use token in the next API request