Policy Configuration

When adding a new policy, you have to configure its details.

The following policies can be found in **Scheer PAS** *API Management*:

Policy Category	Policy Name	Description
Security Policies	**BASIC Authentication Policy**	Enables HTTP BASIC Authentication on an API. Use this policy to require clients to provide HTTP BASIC authentication credentials when making requests to the managed API.
	CORS Policy	Use this policy to enable and configure Cross Origin Resource Sharing on an API. This allows to access resources outside the originating domain.
	Authorization Policy	Allows to add a list of fine-grained authorization rules. Use this policy to control precisely who is allowed to access the API.
	Header Allow /Deny Policy	Allows you to control which incoming requests may be forwarded to the backend service. Permission is granted by adding values for a header.
	HTTP Security Policy	Enforces transport security when using HTTP to mitigate a range of common web vulnerabilities. Contains also a sophisticated mechanism to precisely define the types and sources of content that may be loaded, with violation reporting and the ability to restrict the availability and scope of many security-sensitive features.
	Ignored Resources Policy	Enables the user to shield some API's resources from being accessed, without blocking access to all the API's resources. Requests made to API resources designated as **ignored** result in an HTTP 404 (not found) error code. This policy allows fine-grained control over which of an API's resources are accessible.
	IP Blocklist Policy	This policy blocks access to an API's resource based on the IP address of the client. The user must specify the IP address ranges to be excluded from being able to access the API. Any addresses that are not explicitly excluded are able to access the API. It is possible to use wildcard characters to specify the IP addresses to be blocked. It is also possible to define the return error code sent in the response to the client in case a request is denied. ⚠ An IP Blocklist policy overrides an IP Allowlist policy.
	IP Allowlist Policy	Only inbound API requests from clients, policies, or APIs that satisfy the policy are accepted. The user must specify the IP address ranges to be included to be able to access the API. Any addresses that are not explicitly included are not able to access the API.
	JWT Policy	This policy can set headers as claim values or whole access token.
	Keycloak OAuth Policy	This Keycloak-specific OAuth2 policy regulates access to APIs. It enables a wide range of sophisticated auth facilities in combination with, for instance, Keycloak's federation, brokering and user management capabilities. Keycloak's token format and auth mechanism facilitate excellent performance characteristics, with users able to easily tune the setup to meet their security requirements. ✓ In general, this is one of the best approaches for achieving security without greatly impacting performance.

Related Pages:

- Assigning Policies

	SOAP Authorization Policy	Nearly identical to the Authorization Policy , with the exception that it accepts a SOAP action in the HTTP header. ⚠ This policy will only accept a single SOAP action header. It will not extract the operation name from the SOAP body.
	Time Restricted Access Policy	Manages a list of API routes that can be accessed at specific time and date. This policy allows to control **when** client and users are allowed to access your API.
Limiting Policies	**Rate Limiting Policy**	Governs the number of times requests are made to an API within a specified time period. The requests can be filtered by user, client or API and can set the level of granularity for the time period to second, minute, hour, day, month, or year. The intended use of this policy type is for fine grained processing.
	Transfer Quota Policy	Tracks the number of bytes transferred. Enables the user to set a transfer quota (data) in B, KB, MB or GB for upload, download or both per client, user or API in a definable period of time. The response header can also be freely defined.
Modification Policies	**JSONP Policy**	Turns a standard REST endpoint into a JSONP compatible endpoint. The caller must provide a JSONP callback function name via the URL. ⚠ If the API client does not send the JSONP callback function name in the URL, this policy will do nothing. This allows managed endpoints to support both standard REST **and** JSONP at the same time.
	Simple Header Policy	Headers can be set and removed on request, response or both. The values can be literal strings, environment or system properties. Headers can be removed by simple string equality or regular expression.
	URL Rewriting Policy	This policy is used to re-write responses from the back-end API. They will be modified by fixing up any incorrect URLs found with modified ones. As **Scheer PAS** *API Management* works through an API gateway, an API might return URLs to follow up action or data endpoints. In these cases the back-end API will likely be configured to return a URL pointing to the unmanaged API endpoint. This policy can fix up those URL references so that they point to the managed API endpoint instead.
Other Policies	**APIKey Policy**	Passes the API Key through to the back-end service by adding it to a customizable HTTP header.
	Caching Resources	With this policy it is possible to cache requests based on their URL path, HTTP method and HTTP status code. This allows reducing overall traffic to the backend API.
	Timeout Policy	Allows you to determine timeouts for your API. You can differentiate between a timeout for the initial connection and a timeout for the entire request.