

# Property Manager API (PAPI)



The Property Manager API lets you manage the hostnames and rules that control how Akamai serves your content. Some key new features available via PAPI include **Fast Activation** and **Variables**.

See <https://developer.akamai.com/api/luna/papi/overview.html>

## Variables

Many behavior and criteria options allow you to inject variable text that interprets at runtime on edge servers. PAPI lets you reference a set of built-in system variables as well as your own set of variables based on various inputs.

- Invoke built-in variables:

```
"behaviors": [
  {
    "name": "failAction",
    "options": {
      "netStoragePath": "/path/to/{builtin.
AK_EXTENSION}}/filetype.html",
      ...
    }
  }
]
```

- Variables allow you to parameterize your rules and implement functionality that previously was only possible by injecting advanced XML metadata.
- Built-in variables include request URL details (hostname, path, extension, query strings), client IP, and whether the request was cacheable.

## Combine, Create, and Modify Your Own Variables

1. Extract and modify request context data:

```
{
  "name": "setVariable",
  "options": {
    "variableName": "PMUSER_LANG",
    "valueSource": "EXTRACT",
    "extractLocation": "CLIENT_REQUEST_
HEADER",
    "headerName": "Accept-Language",
    "transform": "UPPER"
  }
}
```

2. Invoke your modified variable elsewhere in the property's rules:

```
"options": {
  "netStoragePath": "/path/to/{user.PMUSER_
LANG}}/language.html",
  ...
}
```

**TIP:** You can also reassign and combine built-in variables and modify them along the way. This makes it much easier to customize your Property Manager rules.

## More Info

<https://developer.akamai.com/api/luna/papi/rules.html#insertingvariables>  
<https://developer.akamai.com/api/luna/papi/behaviors.html#setvariable>

## Automated Akamai Configuration Management

The Property Manager API (PAPI) lets you manage the hostnames and rules that control how Akamai serves your content. Automate common configuration tasks such as creating and cloning configurations, creating and assigning edge hostnames and CP Codes, modifying the rules for serving your content, and activating your configurations on the Akamai staging and production networks.

See <https://developer.akamai.com/api/luna/papi/overview.html>

## Setup

You need the right set of contextual contract, group, and product for all your requests:

1. **GET** `/papi/v1/contracts/` to list contracts, then pick the **contractId** you're working under, e.g. **ctr\_1-1TJZFW**.
2. **GET** `/papi/v1/groups/` to list groups, then pick the **groupId** where you keep the properties you want, e.g. **grp\_15166**.
3. **GET** `/papi/v1/products/?contractId=ctr_1-1TJZFW` to list products available under your contract, then pick the **productId** you want to assign to new properties and Content Provider (CP) codes, e.g. **prd\_Web\_App\_Accel**.

Where it says `{?contractId,groupId}` below, that means a longer query string like `?contractId=ctr_1-1TJZFW&groupId=grp_15166`.

## Create an Edge Hostname

This sets up a new edge hostname and makes it available across the DNS network:

1. Create a request for a new edge hostname.

```
{
  "productId": "prd_Web_App_Accel",
  "domainPrefix": "www.mydomain.com",
  "domainSuffix": "edgesuite.net",
  "secure": true,
  "ipVersionBehavior": "IPV4"
}
```

2. **POST** `/papi/v1/edgehostnames/{?contractId,groupId}`.
3. **GET** the response's `edgeHostnameLink` and store the generated `edgeHostnameId`, such as `ehn_895822`.

The edge hostname takes a few minutes to activate across the DNS network, so don't apply a CNAME a live site to it until its **status** is **ACTIVE**.

## Create a CP Code

CP codes allow you to customize traffic and usage reporting, log delivery, and alerts.

1. Create a request for a new CP code:

```
{
  "productId" : "prd_Web_App_Accel",
  "cpcodeName" : "MyDomain"
}
```

2. **POST** to `/papi/v1/cpcodes/{?contractId,groupId}`
3. **GET** the `cpcodeLink` and store the `cpcodeId` and `cpcodeName`

## Clone an Existing Property

You can maintain a single property version as a template with a basic set of rules to modify. Otherwise, these first two steps let you select one to clone.

1. **GET** `/papi/v1/properties/{?contractId,groupId}` and pick a `propertyId`, such as `prp_175780.sdsd`
2. **GET** `/papi/v1/properties/prp_175780/versions/{?contractId,groupId}` and pick a version number.
3. Create a request for a new property that's cloned from another.

```
{
  "productId": "prd_Obj_Caching",
  "propertyName": "MyDomain"
  "cloneFrom": {
    "propertyId": "prp_175780",
    "version": 5,
    "copyHostnames": false
  }
}
```

4. **POST** it to `/papi/v1/properties/{?contractId,groupId}`.
5. **GET** the response's `propertyLink` and store the new `propertyId`

## Apply a New Hostname

You've already activated the hostname on the edge, but still need to assign it to a specific property.

1. Once you have created an edge hostname you can assign a hostname (digital property) to your property configuration.

```
[
  {
    "cnameType": "EDGE_HOSTNAME",
    "edgeHostnameId": "ehn_895822",
    "cnameFrom": "www.mydomain.com",
```

```
    "cnameTo": "www.mydomain.com.edgesuite.net"
  },
]
```

2. **PUT** it to `/papi/v1/properties/prp_175780/versions/5/hostnames/{?contractId,groupId}`

## Customize the Rule

A property's set of rules is a JSON structure of behavior and criteria objects that execute at the edge. You may need to modify these rules in various ways. One way might be to simply reassign the CP code.

1. **GET** `/papi/v1/properties/prp_175780/versions/5/rules/{?contractId,groupId}`
2. Strip the `cpc_` prefix from your new `cpcodeId` string and convert to numeric, then plug it along with the `cpcodeName` into the top-level `default` rule's `cpCode` behavior:

```
{
  "rules": {
    "name": "default",
    "behaviors": [
      ...
      {
        "name": "cpCode",
        "options": {
          "value": {
            "id" : 12345,
            "name" : "MyDomain"
          }
        }
      }
    ]
  }
}
```

3. **PUT** the object back to the same URL.

## Activate the Property

1. Create an activation request.

```
{
  "propertyVersion": 5,
  "network": "STAGING",
  "note": "test MyDomain on STAGING network",
  "notifyEmails": [ "you@example.com", "them@example.com" ]
}
```

2. **POST** it to `/papi/v1/properties/prp_175780/activations/{?contractId,groupId}`.
3. **GET** the response's `activationLink`. When its status is **ACTIVE**, you're ready to test.

Once you've tested the site on staging, repeat the activation, but with the **network** set to **PRODUCTION**. If you have **Fast Activation** enabled, that should only take about 15 minutes!