



Mitchell Lewars <mitchelllewars@pingidentity.com>

Code for gateway

1 message

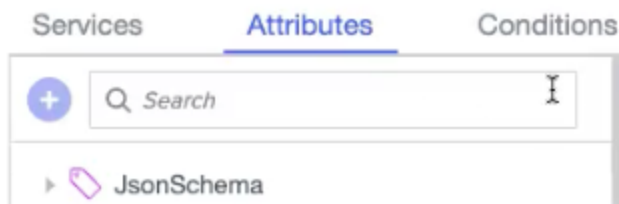
Mitchell Lewars <mitchelllewars@pingidentity.com>
To: Mitchell Lewars <mlewars@pingidentity.com>

Thu, Jul 25, 2024 at 12:31 PM

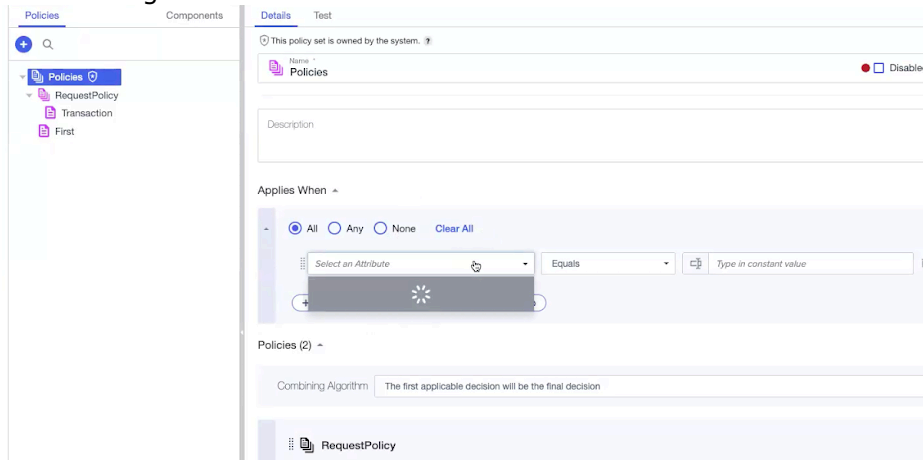
https://pingidentity.zoom.us/rec/share/EQNZTweau9vYJBFEvwHsqckQ5I_k8JTkQybBo4D30UrsZCfTpC5Y4BGMRhoB5J_v.aUQmhrouBuEQCAeE?startTime=17056763180000Y!9xr*Q

<https://drive.google.com/file/d/1Y6SMNZwa8gVjhwtp2X9tus6Tdj34mtfG/view>

1 - Tree Search

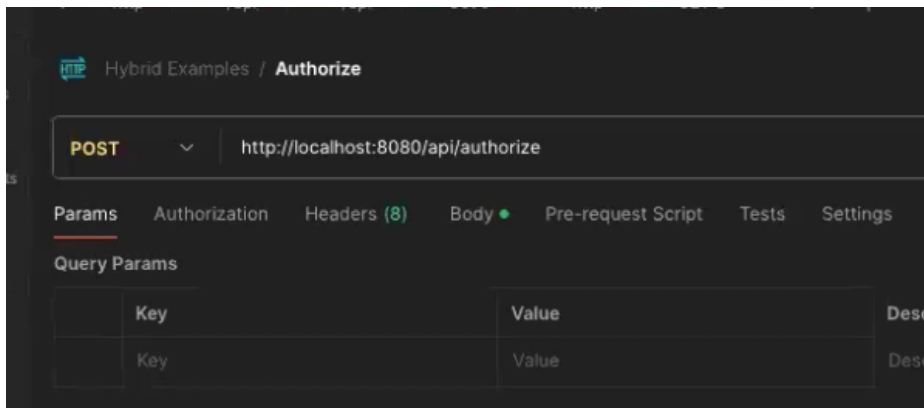


2 - Creating an attribute stub



Click then add new attribute from tree, use name - type
Then select it in the drop-down; new attribute has no resolver in the stub.

3 - Authorize - Version - Endpoint > pick gateway and publish to specific gateway
works with PDP not sideband!

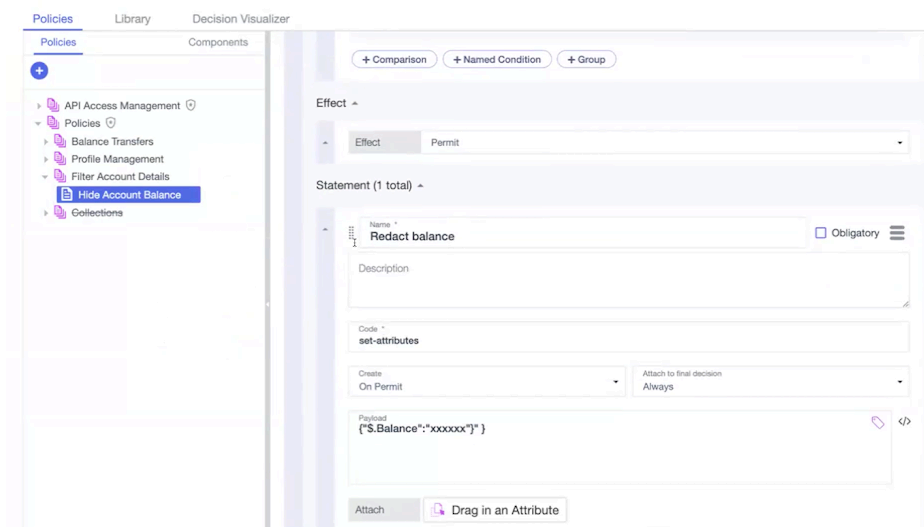
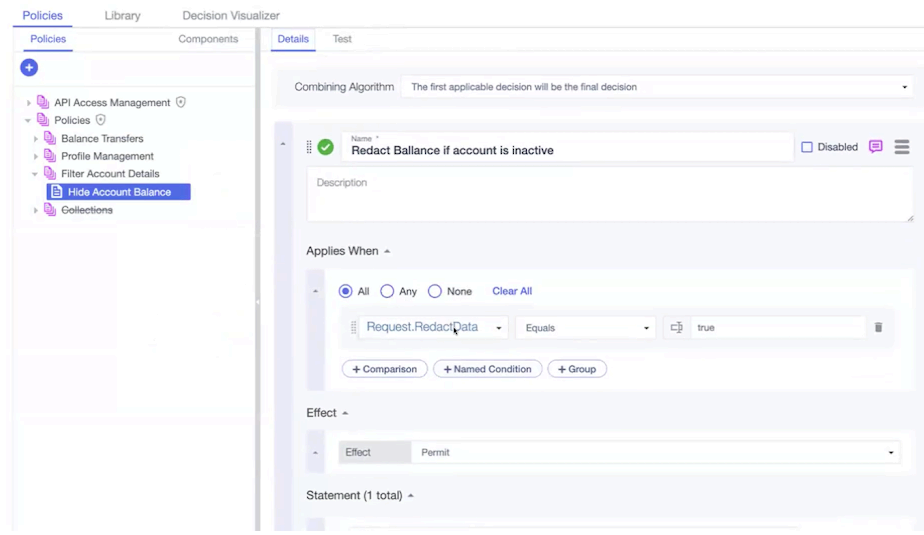


PDP to the gateway instead of calls to PingOne.

PDP Delivered as a lightning gateway...

4 - Using Authorize in a Hybrid - Create local Gateway

5 - Redact Policy -



Allows test and see redact:

Details **Test**

Attributes

Request.User ID 2658

Request.Redact... false

Select an attribute to add it to the testing scenario

Services

Select a service to add it to the testing scenario

☒ Process Statements RESPONSE

API Request

HTTP Method GET

Status * 200

Headers

+ Header

Body

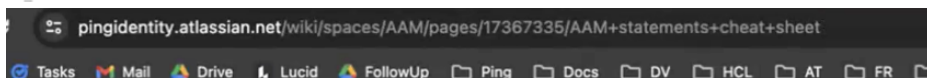
```
{
  "Id": "2658",
  "Forename": "Graham",
  "Surname": "Davidson",
  "Balance": "1000.00"
}
```

Unmodified:

Visualisation Request Response Output Attributes Services **Processing Result**

Body

```
{
  "Id": "2658",
  "Forename": "Graham",
  "Surname": "Davidson",
  "Balance": "1000.00"
}
```



But TRUE:
will show:

Name: Hide Account Balance

Testing Scenario Test Results (1) × Test Results (2) ×

Visualisation Request **Response** Output Attributes Services Processing Result

```

"attributeValueOverrides": {
  "Request.User ID": "2658",
  "Request.RedactData": "true"
},
"serviceValueOverrides": {},
"httpProcessingRequest": {
  "httpFlowType": "RESPONSE",
  "method": "GET",
  "headers": [],
  "statusCode": "OK",
  "body": "{\n  \"Id\": \"2658\", \n  \"Forename\": \"Graham\", \n  \"Surname\": \"Davidson\", \n  \"Balance\": \"xxxxxx\" \n}"
},
"authorized": false,
"statements": [
  {
    "id": "6f573612-465f-4fa4-9f21-8bcb6b5c2a1a",
    "name": "Redact balance",
    "code": "set-attributes",
    "payload": "{ \"S.Balance\": \"xxxxxx\" }",
    "obligatory": false,
    "fulfilled": false,
    "attributes": {}
  }
],
"status": {
  "code": "OKAY",
  "message": "OKAY"
}

```

Testing Scenario Test Results (1) × Test Results (2) ×

Visualisation Request Response **Output** Attributes Services **Processing Result**

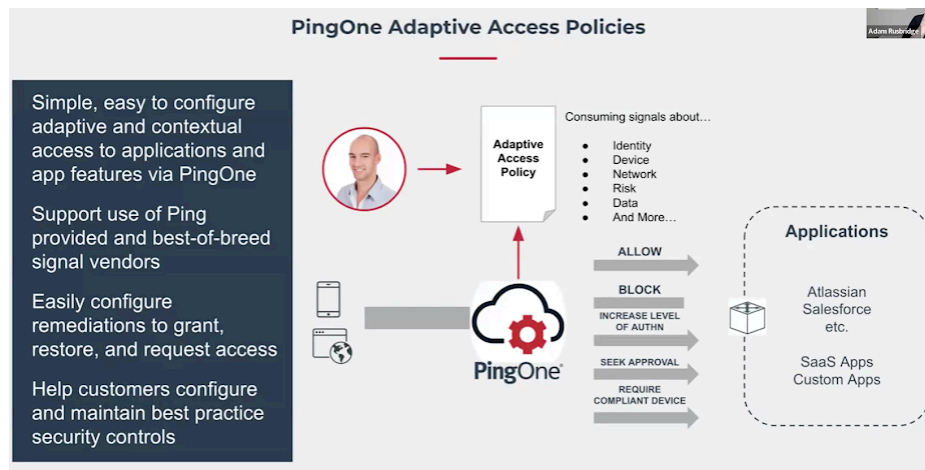
Body

```

{
  "Id": "2658",
  "Forename": "Graham",
  "Surname": "Davidson",
  "Balance": "xxxxxx"
}

```

FEATURE TO COME:



Adaptive Access Policy - Initial Scope

The screenshot shows the 'Adaptive Trust Policies' configuration page in the PingOne console. The left sidebar contains navigation links for Identity, Applications, Dashboard, Authentication, and Integrations. The main panel displays the 'New Policy' form for an 'Adaptive Trust Policy'. The form includes sections for 'User Resource Rules', 'Device Resource Rules', 'Custom Rules', and 'Mitigations and Remediations'. Annotations with blue dots and lines point to specific features: 1. 'Create one or more adaptive access policies and assign to Applications' points to the 'New Policy' button. 2. 'Categorised rules with contextually relevant conditions' points to the 'User Resource Rules' section. 3. 'Integrations with PingOne Protect' points to the 'Device Resource Rules' section. 4. 'Integrations with MDM vendors' points to the 'Custom Rules' section. 5. 'Extensibility with custom attributes and conditions, leveraging 3rd party data sources' points to the 'Custom Rules' section. 6. 'Pre-defined mitigations and Universal Service integrations (MFA, Verify)' points to the 'Mitigations and Remediations' section. 7. 'Exposes a Policy API: Query and response via a policy decision endpoint' points to the 'Policy API' section.

- Create one or more adaptive access policies and assign to Applications
- Categorised rules with contextually relevant conditions
- Integrations with PingOne Protect
- Integrations with MDM vendors
- Extensibility with custom attributes and conditions, leveraging 3rd party data sources
- Pre-defined mitigations and Universal Service integrations (MFA, Verify)
- Exposes a Policy API: Query and response via a policy decision endpoint