Maintenance Guide

ON THIS PAGE

Maintenance Guide

This guide describes how to perform recurring administrative operations in ForgeRock Access Management Java Agent.

About ForgeRock Identity Platform™ Software

ForgeRock® Identity Platform serves as the basis for our simple and comprehensive Identity and Access Management solution. We help our customers deepen their relationships with their customers, and improve the productivity and connectivity of their employees and partners. For more information about ForgeRock and about the platform, see https://www.forgerock.com.

Audit Java Agent

Java Agent logs audit events for security, troubleshooting, and regulatory compliance. Store agent audit event logs in the following ways:

Remotely

Log audit events to the audit event handler configured in the AM realm. In an environment with several AM servers, agents write audit logs to the AM server that satisfies the agent request for client authentication or resource authorization.

Java Agent cannot log audit events remotely if:

- AM's Audit Logging Service is disabled.
- No audit event handler is configured in the realm where the agent is configured.
- All audit event handlers configured in the realm where the agent is configured are disabled.

For more information about audit logging in AM, see Setting Up Audit Logging in AM's *Security Guide*.

Locally

Log audit events in JSON format to a file in the agent installation directory, `/java_agents/agent_type/logs/audit/` .

Remotely and locally

Log audit events:

- To a file in the agent installation directory.
- To the audit event handler configured in the AM realm in which the agent profile is configured.

The following is an example of an agent log record:


```
{
   "timestamp":"...",
   "eventName":"AM-ACCESS-OUTCOME",
   "transactionId":"608831c4-7351-4277-8a5f-b1a83fe2277e",
   "userId":"id=demo,ou=user,dc=openam,dc=forgerock,dc=org",
   "trackingIds":[
      "fd5c8ccf-7d97-49ba-a775-76c3c06eb933-82095",
      "fd5c8ccf-7d97-49ba-a775-76c3c06eb933-82177"
   ],
   "component":"Java Policy Agent",
   "realm":"/",
   "server":{
      "ip":"127.0.0.1",
      "port":8020
   },
   "client":{
      "ip":"127.0.0.1",
      "port":55180
   },
   "request":{
```

```
      "protocol":"HTTP/1.1",
      "operation":"GET"
   },
   "http":{
      "request":{
         "secure":false,
         "method":"GET",
         "path":"http://my.example.com:8020/examples/",
         "headers":{
            "referer":[

"https://openam.example.com:8443/openam/oauth2/authorize?
scope[...]"
            ],
            "accept-language":[
               "en,en-US;q=0.8,da;q=0.6,fr;q=0.4"
            ],
            "host":[
               "my.example.com:8020"
            ],
            "upgrade-insecure-requests":[
               "1"
            ],
            "connection":[
               "keep-alive"
            ],
            "cache-control":[
               "max-age=0"
            ],
            "accept-encoding":[
               "gzip, deflate"
            ],
            "user-agent":[
               "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6)
AppleWebKit/537.36 (KHTML, like Gecko)[...]"
            ],
            "accept":[

"text/html,application/xhtml+xml,application/xml;q=0.9,image/webp
,image/apng,*/*;q=0.8"
            ]
         },
         "cookies":{
            "am-auth-jwt":"eyJ0eXAiOiJKV1QiLCJhbGciOi[...]"
            "i18next":"en",
```

```
            "amlbcookie":"01",
            "iPlanetDirectoryPro":"Ts2zDkGUqgtkoxR[...]"
         }
      }
   },
   "response":{
      "status":"DENIED"
   },
   "_id":"fd5c8ccf-7d97-49ba-a775-76c3c06eb933-81703"
}
```

NOTE

Local audit logs do not have an `_id` attribute, which is an internal AM id.

The audit log format adheres to the log structure shared across the ForgeRock Identity Platform. For more information about the audit log format, see Audit Log Format in AM's *Security Guide*.

Java Agent supports propagation of the transaction ID across the ForgeRock platform, using the HTTP header `X-ForgeRock-TransactionId`. For more information about configuring the header, see Configuring the Trust Transaction Header System Property in AM's *Security Guide*.

Configure Audit Logging

By default, Java Agent does not write audit log records. To configure audit logging, perform this procedure. The agent in this example is in remote configuration mode.

1. In the AM console, select REALMS > Realm Name > Applications > Agents > Java > Agent Name > **Global** > **Audit**.
2. In Audit Access Types, select the type of messages to log. For example, select `LOG_ALL` to log access allowed and access denied events.
3. In Audit Log Location, select whether to write the audit logs locally to the agent installation (`LOCAL`), remotely to AM (`REMOTE`), or to both places (`ALL`). For example, keep `REMOTE` to log audit events to the AM instances.
4. To log audit messages locally, enable Enable Local Audit Log Rotation to rotate the audit log files when they reach a maximum size.
5. If you enabled Enable Local Audit Log Rotation, specify the maximum size of the audit log files in Local Audit Log Rotation Size.

Monitor Performance

Use the following interfaces to monitor Java Agent performance:

Prometheus
: A third-party software used for gathering and processing monitoring data.

  For information about installing and running Prometheus, see the Prometheus documentation.

  You can configure Java Agent to expose an endpoint which Prometheus scrapes to obtain performance metrics from your protected web applications.

  Configure Prometheus to monitor the metrics endpoint exposed by the agent by using the `prometheus.yml` configuration file. For more, see the Prometheus configuration documentation.

  TIP

  Prometheus provides monitoring and processing for the information provided by Java Agent. For further analysis and visualization use tools such as Grafana to create customized charts and graphs based on the information collected by Prometheus.

  Download example Grafana dashboards from the ForgeRock BackStage website. For more information, see the Grafana website.

  For more information, see Expose an Endpoint for Common REST and Prometheus Metrics.

ForgeRock® Common REST
: You can configure Java Agent to expose an endpoint that allows REST clients to gather metrics about your protected web applications, in JSON format.

  For more information, see Expose an Endpoint for Common REST and Prometheus Metrics.

CSV File-based
: Write metrics to comma-separated value (CSV) files, without exposing an endpoint.

  When enabled, the monitoring `.csv` files are written to the same directory as the agent instance debug files, for example in `/path/to/java_agents/tomcat_agent/Agent_001/logs/debug/`.

  For more information, see Save Metrics to CSV Files.

Expose an Endpoint for Common REST and Prometheus Metrics

Common REST and Prometheus performance metrics are provided by an endpoint configured in the protected web application's `web.xml` file. The endpoint must be accessible to the REST client or Prometheus server that will be making use of the performance data.

1. For each protected web application that will expose metrics, edit the web application's `web.xml` file.

   The following Tomcat example exposes a base endpoint named `/metrics`:

   ```
   <servlet>
      <servlet-name>AgentMonitoring</servlet-name>
      <servlet-
   class>org.forgerock.http.servlet.HttpFrameworkServlet</ser
   vlet-class>
      <init-param>
          <param-name>application-loader</param-name>
          <param-value>guice</param-value>
      </init-param>
   </servlet>
   <servlet-mapping>
       <servlet-name>AgentMonitoring</servlet-name>
       <url-pattern>/metrics/*</url-pattern>
   </servlet-mapping>
   ```

   Choose any name for the exposed base endpoint, but make sure it does not conflict with any of the builtin agent endpoints, for example `/sunwCDSSORedirectURI`.

2. Allow access to the base endpoint used for monitoring web applications protected by the agent by using one of the following methods:

   a. Create a Not Enforced URI rule for the base endpoint.

      The following example rule allows access to the metrics base endpoint:

      ```
      */metrics/*
      ```

   b. Create a Compound Not-Enforced URI and IP rule for the base endpoint.

      A Compound Not-Enforced URI and IP rule can allow access from only the IP addresses of the REST clients or Prometheus server.

      The following example rule allows access to the `/metrics` endpoint for HTTP requests that come from the IP address range from 192.168.1.1 to 192.168.1.3:

```
192.168.1.1-192.168.1.3 | */metrics/*
```

HTTP requests from other IP addresses are not able to access the metrics base endpoint.

c. Create an authorization policy in AM to restrict access to the metrics base endpoint.

Note that the metric base endpoint does not require login credentials. You can use a policy to ensure that requests to the endpoints are authenticated against the AM instance.

For more information, see Configuring Policies in AM's *Authorization Guide*.

3. The Common REST performance monitoring endpoint will now be available in the path used by the protected web application, for example `https://mydomain.example.com/myapp/metrics/crest`.

   Configure your REST clients to access the endpoint to gather performance metric data. If you are protecting the endpoint by using policies in AM, include the relevant credentials.

4. The Prometheus performance monitoring endpoint is available in the path used by the protected web application, for example `https://mydomain.example.com/myapp/metrics/prometheus`.

   Configure your Prometheus server to access the endpoint to gather performance metric data. If you are protecting the endpoint by using policies in AM, include the relevant credentials.

Save Metrics to CSV Files

1. Set Export Monitoring Metrics to CSV, as follows:
   - `true` to configure the agent to write metric information to CSV files.
   - `false` to prevent the agent from writing metric information to CSV files.

Metric Types

Timer Fields

Common REST Fields

Field	Description
`_id`	Metric ID.
`_type`	Metric type.
`count`	Number of events recorded for this metric.
`total`	Sum of the durations recorded for this metric.
`min`	Minimum duration recorded for this metric.
`max`	Maximum duration recorded for this metric.
`mean`	Average duration recorded for this metric.
`stddev`	Standard deviation of durations recorded for this metric.
`duration_units`	Units used for measuring the durations in the metric.
`p50`	50% of the durations recorded are at or below this value.
`p75`	75% of the durations recorded are at or below this value.
`p95`	95% of the durations recorded are at or below this value.
`p98`	98% of the durations recorded are at or below this value.
`p99`	99% of the durations recorded are at or below this value.
`p999`	99.9% of the durations recorded are at or below this value.
`m1_rate`	One-minute average rate.
`m5_rate`	Five-minute average rate.
`m15_rate`	Fifteen-minute average rate.
`mean_rate`	Average rate.
`rate_units`	Units used for measuring the rate of the metric.

NOTE

Duration-based values, such as `min`, `max`, and `p50`, are weighted towards newer data. By representing approximately the last five minutes of data, the timers make it easier to see recent changes in behavior, rather than a uniform average of recordings since the server was started.

The following is an example of the `requests.granted.not-enforced` metric from the Common REST endpoint:


```
{
  "_id" : "requests.granted.not-enforced",
  "_type" : "timer",
  "count" : 486,
  "total" : 80.0,
  "min" : 0.0,
  "max" : 1.0,
  "mean" : 0.1905615495053855,
  "stddev" : 0.39274399467782056,
  "duration_units" : "milliseconds",
  "p50" : 0.0,
  "p75" : 0.0,
  "p95" : 1.0,
  "p98" : 1.0,
  "p99" : 1.0,
  "p999" : 1.0,
  "m1_rate" : 0.1819109974890356,
  "m5_rate" : 0.05433445522996721,
  "m15_rate" : 0.03155662103953588,
  "mean_rate" : 0.020858521722211427,
  "rate_units" : "calls/second"
}
```


Prometheus Fields

The Prometheus endpoint does not provide rate-based statistics, as rates can be calculated from the time-series data.

Field	Description
`# TYPE`	Metric ID, and type. Note that the `Timer` metric type is reported as a `Summary` type. Formatted as a comment.
`_count`	Number of events recorded.
`_total`	Sum of the durations recorded.
`{quantile="0.5"}`	50% of the durations are at or below this value.
`{quantile="0.75"}`	75% of the durations are at or below this value.

Field	Description
`{quantile="0.95"}`	95% of the durations are at or below this value.
`{quantile="0.98"}`	98% of the durations are at or below this value.
`{quantile="0.99"}`	99% of the durations are at or below this value.
`{quantile="0.999"}`	99.9% of the durations are at or below this value.

NOTE

Duration-based quantile values are weighted towards newer data. By representing approximately the last five minutes of data, the timers make it easier to see recent changes in behavior, rather than a uniform average of recordings since the server was started.

The following is an example of the `ja_requests{access=granted,decision=allowed-by-policy}` metric from the Prometheus endpoint:

```
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.5",} 0.013000000000000001
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.75",} 0.022000000000000002
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.95",} 0.022000000000000002
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.98",} 0.022000000000000002
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.99",} 0.022000000000000002
ja_requests_seconds{access="granted",decision="allowed-by-
policy",quantile="0.999",} 1.1380000000000001
ja_requests_count{access="granted",decision="allowed-by-policy",}
7.0
ja_requests_seconds_total{access="granted",decision="allowed-by-
policy",} 1.21
```

Gauge Fields

Common REST Fields

Metric for a numerical value that can increase or decrease. The value for a gauge is calculated when requested, and represents the state of Metric at that specific time.

Field	Description
`_id`	Metric ID.
`_type`	Metric type.
`value`	Current value of the metric.

The following is an example of the `jvm.used-memory` metric from the Common REST endpoint:

```
{
  "_id" : "jvm.used-memory",
  "_type" : "gauge",
  "value" : 2.13385216E9
}
```

Prometheus Fields

Field	Description
`# TYPE`	Metric ID, and type. Formatted as a comment.
`{Metric ID}`	Current value. Large values may be represented in scientific E-notation.

The following is an example of the `ja_jvm_used_memory_bytes` metric from the Prometheus endpoint:

```
# TYPE ja_jvm_used_memory_bytes gauge
ja_jvm_used_memory_bytes 1.418723328E9
```

DistinctCounter

Metric providing an estimate of the number of *unique* values recorded.

For example, this could be used to estimate the number of unique users who have authenticated, or unique client IP addresses.

NOTE

NOTE

The `DistinctCounter` metric is calculated per instance of AM, and cannot be aggregated across multiple instances to get a site-wide view.

Common REST Fields

Field	Description
`_id`	Metric ID.
`_type`	Metric type. Note that the `distinctCounter` type is reported as a `gauge` type. The output formats are identical.
`value`	Calculated estimate of the number of unique values recorded in the metric.

The following is an example of the `authentication.unique-uuid.success` metric from the Common REST endpoint:

```
{
  "_id" : "authentication.unique-uuid.success",
  "_type" : "gauge",
  "value" : 3.0
}
```

Prometheus Fields

Field	Description
`# TYPE`	Metric ID, and type. Note that the `distinctCounter` type is reported as a `gauge` type. The output formats are identical. Formatted as a comment.
`{Metric ID}`	Calculated estimate of the number of unique values recorded in the metric.

The following is an example of the `ja_notenforced_ip_unmatched_cache_size` metric from the Prometheus endpoint:

```
# TYPE ja_notenforced_ip_unmatched_cache_size gauge
ja_notenforced_ip_unmatched_cache_size 3.0
```

Exposed Metrics

Java Agent exposes the monitoring metrics described in this section.

Audit Handler Metrics

Metric	Prometheus name	Description
`audit.access.generate`	`ja_audit_generate{topic=access}`	Time taken to generate an audit object. (Timer)
`audit.handler.<handler-type>.default.access.<outcome>`	`ja_audit{handler-type=<handler-type>,name=default,topic=access,outcome=<outcome>}`	Time taken to audit outcomes, both locally to the agent and remotely in AM. (Timer)

Labels:

<handler-type>

`am-delegate` . Remote auditing performed by AM. (Prometheus: `am_delegate`)

`json` . Local audit logging using JSON.

<outcome>

`success`

`failure`

Endpoint and REST SDK Metrics

Metric	Prometheus name	Description
`session-info`	`ja_session_info`	Time taken to retrieve user session information from AM. (Timer)
`user-profile`	`ja_user_profile`	Time taken to retrieve the user profile information from AM. (Timer)
`policy-decision`	`ja_policy_decision`	Time taken to retrieve policy decisions from AM. (Timer)

JSON Web Token (JWT) Metrics

Metric	Prometheus name	Description
`jwt.cache.size`	`ja_jwt_cache_size`	
Size of the JWT cache. (Gauge)	`jwt.cache.eviction`	`ja_jwt_cache_eviction`
The eviction count for the JWT cache. (Gauge)	`jwt.cache.load-count`	`ja_jwt_cache_load_count`
The load count for the JWT cache. (Gauge)	`jwt.cache.load-time`	`ja_jwt_cache_load_time`
The load time for the JWT cache, in milliseconds. (Gauge)	`jwt.cache.hit`	`ja_jwt_cache{outcome=hit}`
The hit count for the JWT cache. (Gauge)	`jwt.cache.miss`	`ja_jwt_cache{outcome=miss}`

JVM Metrics

TIP

To get Metric name used by Prometheus, prepend `ja_` to the names below, and replace period (`.`) and hyphen (`-`) characters with underscore (`_`) characters. For example, the `jvm.available-cpus` metric is named `ja_jvm_available_cpus` in Prometheus.

Name	Description
`jvm.available-cpus`	Number of processors available to the Java virtual machine. (Gauge)
`jvm.class-loading.loaded`	Number of classes loaded since the Java virtual machine started. (Gauge)
`jvm.class-loading.unloaded`	Number of classes unloaded since the Java virtual machine started. (Gauge)
`jvm.garbage-collector.PS-MarkSweep.count`	Number of collections performed by the "parallel scavenge mark sweep" garbage collection algorithm. (Gauge)
`jvm.garbage-collector.PS-MarkSweep.time`	Approximate accumulated time taken by the "parallel scavenge mark sweep" garbage collection algorithm. (Gauge)

Name	Description
`jvm.garbage-collector.PS-Scavenge.count`	Number of collections performed by the "parallel scavenge" garbage collection algorithm. (Gauge)
`jvm.garbage-collector.PS-Scavenge.time`	Approximate accumulated time taken by the "parallel scavenge" garbage collection algorithm. (Gauge)
`jvm.memory-usage.heap.init`	Amount of heap memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.heap.max`	Maximum amount of heap memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.heap.committed`	Amount of heap memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.heap.used`	Amount of heap memory used by the Java virtual machine. (Gauge)
`jvm.memory-usage.total.init`	Amount of memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.total.max`	Maximum amount of memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.non-heap.init`	Amount of non-heap memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.non-heap.max`	Maximum amount of non-heap memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.non-heap.committed`	Amount of non-heap memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.non-heap.used`	Amount of non-heap memory used by the Java virtual machine. (Gauge)

Name	Description
`jvm.memory-usage.pools.Code-Cache.init`	Amount of "code cache" memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.pools.Code-Cache.max`	Maximum amount of "code cache" memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.pools.Code-Cache.committed`	Amount of "code cache" memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.pools.Code-Cache.used`	Amount of "code cache" memory used by the Java virtual machine. (Gauge)
`jvm.memory-usage.pools.Compressed-Class-Space.init`	Amount of "compressed class space" memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.pools.Compressed-Class-Space.init`	Maximum amount of "compressed class space" memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.pools.Compressed-Class-Space.committed`	Amount of "compressed class space" memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.pools.Compressed-Class-Space.used`	Amount of "compressed class space" memory used by the Java virtual machine. (Gauge)
`jvm.memory-usage.pools.Metaspace.init`	Amount of "metaspace" memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.pools.Metaspace.max`	Maximum amount of "metaspace" memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.pools.Metaspace.committed`	Amount of "metaspace" memory that is committed for the Java virtual machine to use. (Gauge)

Name	Description
jvm.memory-usage.pools.Metaspace.used	Amount of "metaspace" memory used by the Java virtual machine. (Gauge)
jvm.memory-usage.pools.PS-Eden-Space.init	Amount of "parallel scavenge eden space" memory that the Java virtual machine initially requested from the operating system. (Gauge)
jvm.memory-usage.pools.PS-Eden-Space.max	Maximum amount of "parallel scavenge eden space" memory that the Java virtual machine will attempt to use. (Gauge)
jvm.memory-usage.pools.PS-Eden-Space.committed	Amount of "parallel scavenge eden space" memory that is committed for the Java virtual machine to use. (Gauge)
jvm.memory-usage.pools.PS-Eden-Space.used-after-gc	Amount of "parallel scavenge eden space" memory after the last time garbage collection recycled unused objects in this memory pool. (Gauge)
jvm.memory-usage.pools.PS-Eden-Space.used	Amount of "parallel scavenge eden space" memory used by the Java virtual machine. (Gauge)
jvm.memory-usage.pools.PS-Old-Gen.init	Amount of "parallel scavenge old generation" memory that the Java virtual machine initially requested from the operating system. (Gauge)
jvm.memory-usage.pools.PS-Old-Gen.max	Maximum amount of "parallel scavenge old generation" memory that the Java virtual machine will attempt to use. (Gauge)
jvm.memory-usage.pools.PS-Old-Gen.committed	Amount of "parallel scavenge old generation" memory that is committed for the Java virtual machine to use. (Gauge)
jvm.memory-usage.pools.PS-Old-Gen.used-after-gc	Amount of "parallel scavenge old generation" memory after the last time garbage collection recycled unused objects in this memory pool. (Gauge)

Name	Description
`jvm.memory-usage.pools.PS-Old-Gen.used`	Amount of "parallel scavenge old generation" memory used by the Java virtual machine. (Gauge)
`jvm.memory-usage.pools.PS-Survivor-Space.init`	Amount of "parallel scavenge survivor space" memory that the Java virtual machine initially requested from the operating system. (Gauge)
`jvm.memory-usage.pools.PS-Survivor-Space.max`	Maximum amount of "parallel scavenge survivor space" memory that the Java virtual machine will attempt to use. (Gauge)
`jvm.memory-usage.pools.PS-Survivor-Space.committed`	Amount of "parallel scavenge survivor space" memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.pools.PS-Survivor-Space.used-after-gc`	Amount of "parallel scavenge survivor space" memory after the last time garbage collection recycled unused objects in this memory pool. (Gauge)
`jvm.memory-usage.pools.PS-Survivor-Space.used`	Amount of "parallel scavenge survivor space" memory used by the Java virtual machine. (Gauge)
`jvm.memory-usage.total.committed`	Amount of memory that is committed for the Java virtual machine to use. (Gauge)
`jvm.memory-usage.total.used`	Amount of memory used by the Java virtual machine. (Gauge)
`jvm.thread-state.blocked.count`	Number of threads in the BLOCKED state. (Gauge)
`jvm.thread-state.count`	Number of live threads including both daemon and non-daemon threads. (Gauge)
`jvm.thread-state.daemon.count`	Number of live daemon threads. (Gauge)
`jvm.thread-state.new.count`	Number of threads in the NEW state. (Gauge)

Name	Description
`jvm.thread-state.runnable.count`	Number of threads in the RUNNABLE state. (Gauge)
`jvm.thread-state.terminated.count`	Number of threads in the TERMINATED state. (Gauge)
`jvm.thread-state.timed_waiting.count`	Number of threads in the TIMED_WAITING state. (Gauge)
`jvm.thread-state.waiting.count`	Number of threads in the WAITING state. (Gauge)

Not Enforced Rule Metrics

Metric	Prometheus name	Description
`notenforced-uri.matched.cache.size`	`ja_notenforced_uri_matched_cache_size`	Size of the not-enforced URI matched cache. (Gauge)
`notenforced-uri.matched.cache.eviction`	`ja_notenforced_uri_matched_cache_eviction`	Eviction count for the not-enforced URI matched cache. (Gauge)
`notenforced-uri.matched.cache.load-count`	`ja_notenforced_uri_matched_cache_load_count`	Load count for the not-enforced URI matched cache. (Gauge)
`notenforced-uri.matched.cache.load-time`	`ja_notenforced_uri_matched_cache_load_time`	Load time for the not-enforced URI matched cache, in milliseconds. (Gauge)
`notenforced-uri.matched.cache.hit`	`ja_notenforced_uri_matched_cache{outcome=hit}`	Hit count for the not-enforced URI matched cache. (Gauge)
`notenforced-uri.matched.cache.miss`	`ja_notenforced_uri_matched_cache{outcome=miss}`	Miss count for the not-enforced URI matched cache. (Gauge)
`notenforced-uri.unmatched.cache.size`	`ja_notenforced_uri_unmatched_cache_size`	Size of the not-enforced URI unmatched cache. (Gauge)

Metric	Prometheus name	Description
`notenforced-uri.unmatched.cache.eviction`	`ja_notenforced_uri_unmatched_cache_eviction`	Eviction count for the not-enforced URI unmatched cache. (Gauge)
`notenforced-uri.unmatched.cache.load-count`	`ja_notenforced_uri_unmatched_cache_load_count`	Load count for the not-enforced URI unmatched cache. (Gauge)
`notenforced-uri.unmatched.cache.load-time`	`ja_notenforced_uri_unmatched_cache_load_time`	Load time for the not-enforced URI unmatched cache, in milliseconds. (Gauge)
`notenforced-uri.unmatched.cache.hit`	`ja_notenforced_uri_unmatched_cache{outcome=hit}`	Hit count for the not-enforced URI unmatched cache. (Gauge)
`notenforced-uri.unmatched.cache.miss`	`ja_notenforced_uri_unmatched_cache{outcome=miss}`	Miss count for the not-enforced URI unmatched cache. (Gauge)
`notenforced-ip.matched.cache.size`	`ja_notenforced_ip_matched_cache_size`	Size of the not-enforced IP matched cache. (Gauge)
`notenforced-ip.matched.cache.eviction`	`ja_notenforced_ip_matched_cache_eviction`	Eviction count for the not-enforced IP matched cache. (Gauge)
`notenforced-ip.matched.cache.load-count`	`ja_notenforced_ip_matched_cache_load_count`	Load count for the not-enforced IP matched cache. (Gauge)
`notenforced-ip.matched.cache.load-time`	`ja_notenforced_ip_matched_cache_load_time`	Load time for the not-enforced IP matched cache, in milliseconds. (Gauge)
`notenforced-ip.matched.cache.hit`	`ja_notenforced_ip_matched_cache{outcome=hit}`	Hit count for the not-enforced IP matched cache. (Gauge)
`notenforced-ip.matched.cache.miss`	`ja_notenforced_ip_matched_cache{outcome=miss}`	Miss count for the not-enforced IP matched cache. (Gauge)

Metric	Prometheus name	Description
notenforced-ip.unmatched.cache.size	ja_notenforced_ip_unmatched_cache_size	Size of the not-enforced IP unmatched cache. (Gauge)
notenforced-ip.unmatched.cache.eviction	ja_notenforced_ip_unmatched_cache_eviction	Eviction count for the not-enforced IP unmatched cache. (Gauge)
notenforced-ip.unmatched.cache.load-count	ja_notenforced_ip_unmatched_cache_load_count	Load count for the not-enforced IP unmatched cache. (Gauge)
notenforced-ip.unmatched.cache.load-time	ja_notenforced_ip_unmatched_cache_load_time	Load time for the not-enforced IP unmatched cache, in milliseconds. (Gauge)
notenforced-ip.unmatched.cache.hit	ja_notenforced_ip_unmatched_cache{outcome=hit}	Hit count for the not-enforced IP unmatched cache. (Gauge)
notenforced-ip.unmatched.cache.miss	ja_notenforced_ip_unmatched_cache{outcome=miss}	Miss count for the not-enforced IP unmatched cache. (Gauge)
notenforced-compound.matched.cache.size	ja_notenforced_compound_matched_cache_size	Size of the not-enforced compound matched cache. (Gauge)
notenforced-compound.matched.cache.eviction	ja_notenforced_compound_matched_cache_eviction	Eviction count for the not-enforced compound matched cache. (Gauge)
notenforced-compound.matched.cache.load-count	ja_notenforced_compound_matched_cache_load_count	Load count for the not-enforced compound matched cache. (Gauge)
notenforced-compound.matched.cache.load-time	ja_notenforced_compound_matched_cache_load_time	Load time for the not-enforced compound matched cache, in milliseconds. (Gauge)
notenforced-compound.matched.cache.hit	ja_notenforced_compound_matched_cache{outcome=hit}	Hit count for the not-enforced compound matched cache. (Gauge)

Metric	Prometheus name	Description
notenforced-compound.matched.cache.miss	ja_notenforced_compound_matched_cache{outcome=miss}	Miss count for the not-enforced compound matched cache. (Gauge)
notenforced-compound.unmatched.cache.size	ja_notenforced_compound_unmatched_cache_size	Size of the not-enforced compound unmatched cache. (Gauge)
notenforced-compound.unmatched.cache.eviction	ja_notenforced_compound_unmatched_cache_eviction	Eviction count for the not-enforced compound unmatched cache. (Gauge)
notenforced-compound.unmatched.cache.load-count	ja_notenforced_compound_unmatched_cache_load_count	Load count for the not-enforced compound unmatched cache. (Gauge)
notenforced-compound.unmatched.cache.load-time	ja_notenforced_compound_unmatched_cache_load_time	Load time for the not-enforced compound unmatched cache, in milliseconds. (Gauge)
notenforced-compound.unmatched.cache.hit	ja_notenforced_compound_unmatched_cache{outcome=hit}	Hit count for the not-enforced compound unmatched cache. (Gauge)
notenforced-compound.unmatched.cache.miss	ja_notenforced_compound_unmatched_cache{outcome=miss}	Miss count for the not-enforced compound unmatched cache. (Gauge)

Policy Decision Metrics

Metric	Prometheus name	Description
policy-decision.cache.size	ja_policy_decision_cache_size	Size of the policy decision cache. (Gauge)
policy-decision.cache.eviction	ja_policy_decision_cache_eviction	Eviction count for the policy decision cache. (Gauge)
policy-decision.cache.load-count	ja_policy_decision_cache_load_count	Load count for the policy decision cache. (Gauge)

Metric	Prometheus name	Description
`policy-decision.cache.load-time`	`ja_policy_decision_cache_load_time`	Load time for the policy decision cache, in milliseconds. (Gauge)
`policy-decision.cache.hit`	`ja_policy_decision_cache{outcome=hit}`	Hit count for the policy decision cache. (Gauge)
`policy-decision.cache.miss`	`ja_policy_decision_cache{outcome=miss}`	Miss count for the policy decision cache. (Gauge)

POST Data Preservation Metrics

Metric	Prometheus name	Description
`pdp.cache.size`	`ja_pdp_cache_size`	Size of the POST data preservation cache. (Gauge)
`pdp.cache.eviction`	`ja_pdp_cache_eviction`	Eviction count for the POST data preservation cache. (Gauge)
`pdp.cache.load-count`	`ja_pdp_cache_load_count`	Load count for the POST data preservation cache. (Gauge)
`pdp.cache.load-time`	`ja_pdp_cache_load_time`	Load time for the POST data preservation cache, in milliseconds. (Gauge)
`pdp.cache.hit`	`ja_pdp_cache{outcome=hit}`	Hit count for the POST data preservation cache. (Gauge)
`pdp.cache.miss`	`ja_pdp_cache{outcome=miss}`	Miss count for the POST data preservation cache. (Gauge)

Request Metrics

Metric	Prometheus name	Description
`requests.<access>.<decision>`	`ja_requests{access=<access>,decision=<decision>}`	Rate of granted/denied requests and their decision. (Timer)

Labels:

<access>
: `granted`

: `denied`

<decision>
: `not-enforced` : Request matched a not enforced rule.

: `no-valid-token` : Request did not have a valid SSO token or an OpenID Connect JWT.

: `allowed-by-policy` : Request matched a policy, which allowed access.

: `denied-by-policy` : Request matched a policy, which denied access.

: `am-unavailable` : The AM instance was not reachable.

: `agent-exception` : An internal error (exception) occurred within the agent.

Session Information Metrics

Metric	Prometheus name	Description
`session-info.cache.size`	`ja_session_info_cache_size`	Size of the session information cache. (Gauge)
`session-info.cache.eviction`	`ja_session_info_cache_eviction`	Eviction count for the session information cache. (Gauge)
`session-info.cache.load-count`	`ja_session_info_cache_load_count`	Load count for the session information cache. (Gauge)
`session-info.cache.load-time`	`ja_session_info_cache_load_time`	Load time for the session information cache, in milliseconds. (Gauge)
`session-info.cache.hit`	`ja_session_info_cache{outcome=hit}`	Hit count for the session information cache. (Gauge)

Metric	Prometheus name	Description
`session-info.cache.miss`	`ja_session_info_cache{outcome=miss}`	Miss count for the session information cache. (Gauge)

SSO Token to JWT Exchange Metrics

Metric	Prometheus name	Description
`sso-exchange.cache.size`	`ja_sso_exchange_cache_size`	Size of the SSO token exchange cache. (Gauge)
`sso-exchange.cache.eviction`	`ja_sso_exchange_cache_eviction`	Eviction count for the SSO token exchange cache. (Gauge)
`sso-exchange.cache.load-count`	`ja_sso_exchange_cache_load_count`	Load count for the SSO token exchange cache. (Gauge)
`sso-exchange.cache.load-time`	`ja_sso_exchange_cache_load_time`	Load time for the SSO token exchange, in milliseconds. (Gauge)
`sso-exchange.cache.hit`	`ja_sso_exchange_cache{outcome=hit}`	Hit count for the SSO token exchange cache. (Gauge)
`sso-exchange.cache.miss`	`ja_sso_exchange_cache{outcome=miss}`	Miss count for the SSO token exchange cache. (Gauge)

Websocket Metrics

Metric	Prometheus name	Description
`websocket.last-received`	`ja_websocket_last_received`	Number of milliseconds since anything was received over the websocket, for example a ping or a notification. (Gauge)

Metric	Prometheus name	Description
`websocket.last-sent`	`ja_websocket_last_sent`	Number of milliseconds since anything was sent over the websocket. (Gauge)
`websocket.config-change.received`	`ja_websocket_config_change_received`	Number of configuration change notifications received. Note that some may be ignored if the realm or agent name are not applicable. (DistinctCounter)
`websocket.config-change.processed`	`ja_websocket_config_change_processed`	Number of configuration change notifications processed, that were not ignored. (DistinctCounter)
`websocket.policy-change.received`	`ja_websocket_policy_change_received`	Number of policy change notifications received. Note that some may be ignored if the realm or agent name are not applicable. (DistinctCounter)
`websocket.policy-change.processed`	`ja_websocket_policy_change_processed`	Number of policy change notifications processed, that were not ignored. (DistinctCounter)
`websocket.session-logout.received`	`ja_websocket_session_logout_received`	Number of session logout notifications received. Note that some may be ignored if the realm or agent name are not applicable. (DistinctCounter)
`websocket.session-logout.processed`	`ja_websocket_session_logout_processed`	Number of session logout notifications processed, that were not ignored. (DistinctCounter)
`websocket.ping-pong`	`ja_websocket_ping_pong`	Ping/pong round trip time. (Timer)

Notifications

AM sends the following notifications to Java Agent through WebSockets:

Configuration notifications
: When the administrator makes a change to an agent configuration property, AM sends a notification to the agent. The agent flushes the configuration cache, and rereads the agent profile from AM. For more information about the cache, see Configuration Cache.

  Configuration notifications apply when you store the agent profile in AM's configuration data store. For information, see Enable Notifications of Agent Configuration Change.

Session notifications
: When a client logs out, or a CTS-based session expires, AM sends a notification to the agent to remove that entry from the session cache. For information, see Enable Notification of Session Logout. For more information about the cache, see Session Cache.

Policy notifications
: When an administrator changes a policy, AM sends a notification to the agent to flush the policy cache. For information, see Enable Notification of Policy Changes. For more information about the cache, see Policy Cache.

Notifications are enabled by default. To disable notifications, unsubscribe separately to each type of notification. If Autonomous mode is `true`, notifications and many other features are automatically disabled.

In configurations with load balancers and reverse proxies, make sure that the load balancers and reverse proxies support WebSockets.

For more information about properties that configure notifications, see Notifications in the *Properties Reference*.