OpenAM Web Policy Agent 3.3 Reference

Mark Craig
Vanessa Richie
Mike Jang
Chris Lee

,
, ,

Abstract

Guide to installing OpenAM web policy agents. OpenAM provides open source Authentication, Authorization, Entitlement and Federation software.

Table of Contents

Chapter 1

Web Agent Configuration Properties

Web agents use the following configuration properties. Bootstrap properties are always configured locally, whereas other agent configuration properties are either configured centrally in OpenAM or locally using the agent properties file.

1.1. Bootstrap Configuration Properties

These properties are set in `config/`.

`com.forgerock.agents.ext.url.validation.default.url.set`

This property takes a comma-separated list of indexes for URL values indicating the order in which to fail over, where the indexes are taken from the values set for `com.sun.identity.agents.config.naming.url`, `com.sun.identity.agents.config.login.url`, `com.sun.identity.agents.config.cdsso.cdcservlet.url`, and `com.sun.identity.agents.config.logout.url`.

For example if `com.sun.identity.agents.config.naming.url` is set as follows:

```
com.sun.identity.agents.config.naming.url=
 http://zero.example.com:8080/openam/namingservice
 http://one.example.com:8080/openam/namingservice
```

Then the following setting means first use OpenAM on `zero.example.com`, then fail over if necessary to OpenAM on `one.example.com`, assuming `com.forgerock.agents.ext.url.validation.level` is set to enable validation.

```
com.forgerock.agents.ext.url.validation.default.url.set=0,1
```

When using this failover capability make sure you synchronize URL settings in `com.sun.identity.agents.config.naming.url`, `com.sun.identity.agents.config.login.url`, `com.sun.identity.agents.config.cdsso.cdcservlet.url`, and `com.sun.identity.agents.config.logout.url` such that each service shares the same index across all properties. In other words, in the example above each service under `http://zero.example.com:8080/openam` would be the first item (index: 0) for each property. This ensures the policy agent fails over and fails back from one server to another in synchronized fashion for all services.

This property has no default setting.

com.forgerock.agents.ext.url.validation.level

This bootstrap configuration property lets you configure naming URL validation during the initial bootstrap phase when the policy agent reads its configuration, and then thereafter if the policy agent is configured fail over when a naming URL becomes invalid.

When URL validation is fully disabled the policy agent does not need to connect to OpenAM during the bootstrap phase.

If you leave naming URL validation disabled, then make sure that the URLs in the policy agent bootstrap configuration file are valid and correct. As the policy agent performs no further validation after the bootstrap phase, incorrect naming URLs can cause the agent to crash.

To enable full URL validation, set the property as shown:

```
com.forgerock.agents.ext.url.validation.level = 0
```

This property can take the following values.

0

Fully validate naming URLs specified by using the `com.sun.identity.agents.config.naming.url` property. The web policy agent logs into and logs out of OpenAM to check that a naming URL is valid.

1

Check that naming URLs are valid by performing an HTTP GET, which should receive an HTTP 200 response.

2 (Default)

Disable all naming URL validation.

When naming URL validation is enabled, then set the following properties.

- `com.sun.identity.agents.config.connect.timeout`
- `com.sun.identity.agents.config.receive.timeout`

com.forgerock.agents.ext.url.validation.ping.interval

Set this to the seconds between validation requests against the current naming URL.

The sum of the values of `com.sun.identity.agents.config.connect.timeout` and `com.sun.identity.agents.config.receive.timeout` must not exceed this value. Notice that the two timeout values are specified in milliseconds, whereas this property's value is specified in seconds.

Default: 60 (seconds)

`com.forgerock.agents.ext.url.validation.ping.miss.count`

If validation requests against the current naming URL fail this number of times in a row, the web policy agent fails over to the next service in `com.forgerock.agents.ext.url.validation.default.url.set`.

Default: 3

`com.forgerock.agents.ext.url.validation.ping.ok.count`

After failover, if validation requests against the default naming URL succeed this number of times in a row, the web policy agent fails back to that service, the first URL in the `com.forgerock.agents.ext.url.validation.default.url.set` list.

Default: 3

`com.sun.identity.agents.config.certdb.password`

When SSL is configured, set this to the password for the certificate database.

`com.sun.identity.agents.config.certdb.prefix`

When SSL is configured, set this property if the certificate databases in the directory specified by `com.sun.identity.agents.config.sslcert.dir` have a prefix.

`com.sun.identity.agents.config.certificate.alias`

When SSL is configured, set this to the alias of the certificate used to authenticate.

`com.sun.identity.agents.config.connect.timeout`

Set this to the number of milliseconds to keep the socket connection open before timing out. If you have the web policy agent perform naming URL validation, then set this property to a reasonable value such as 2000 (2 seconds). The default value is 0 which implies no timeout.

`com.sun.identity.agents.config.debug.file`

Set this to the full path of the agent's debug log file.

`com.sun.identity.agents.config.debug.level`

Default is `Error`. Increase to `Message` or even `All` for fine-grained detail.

Set the level in the configuration file by module using the format *`module[:level][,module[:level]]*`*, where *module* is one of `AuthService`, `NamingService`, `PolicyService`, `SessionService`, `PolicyEngine`, `ServiceEngine`, `Notification`, `PolicyAgent`, `RemoteLog`, or `all`, and *level* is one of the following.

- `0`: Disable logging from specified module

  At this level the agent nevertheless logs messages having the level value `always`.

- `1`: Log error messages
- `2`: Log warning and error messages

- `3`: Log info, warning, and error messages
- `4`: Log debug, info, warning, and error messages
- `5`: Like level 4, but with even more debugging messages

When you omit *level*, the agent uses the default level, which is the level associated with the `all` module.

The following example used in the local configuration sets the log overall level to debug for all messages.

```
com.sun.identity.agents.config.debug.level=all:4
```

`com.sun.identity.agents.config.forward.proxy.host`

When OpenAM and the agent communicate through a web proxy server configured in forward proxy mode, set this to the proxy server host name.

`com.sun.identity.agents.config.forward.proxy.password`

When OpenAM and the agent communicate through a web proxy server configured in forward proxy mode and the proxy server has the agent authenticate using Basic Authentication, set this to the agent's password.

`com.sun.identity.agents.config.forward.proxy.port`

When OpenAM and the agent communicate through a web proxy server configured in forward proxy mode, set this to the proxy server port number.

`com.sun.identity.agents.config.forward.proxy.user`

When OpenAM and the agent communicate through a web proxy server configured in forward proxy mode and the proxy server has the agent authenticate using Basic Authentication, set this to the agent's user name.

`com.sun.identity.agents.config.key`

Set this to the encryption key used to encrypt the agent profile password.

`com.sun.identity.agents.config.local.logfile`

Set this to the full path for agent's audit log file.

`com.sun.identity.agents.config.naming.url`

Set this to the naming service URL(s) used for naming lookups in OpenAM. Separate multiple URLs with single space characters.

`com.sun.identity.agents.config.organization.name`

Set this to the realm name where the agent authenticates to OpenAM.

`com.sun.identity.agents.config.password`

Set this to the encrypted version of the password for the agent authenticator. Use the command **./agentadmin --encrypt *agentInstance passwordFile*** to get the encrypted version.

`com.sun.identity.agents.config.profilename`

Set this to the agent profile name.

`com.sun.identity.agents.config.receive.timeout`

Set this to the number of milliseconds to wait for a response from OpenAM before timing out and dropping the connection. If you have the web policy agent perform naming URL validation, then set this property to a reasonable value such as 2000 (2 seconds). The default value is 0 which implies no timeout.

`com.sun.identity.agents.config.sslcert.dir`

When SSL is configured, set this to the directory containing SSL certificate databases.

`com.sun.identity.agents.config.tcp.nodelay.enable`

Set to `true` to enable the socket option `TCP_NODELAY`. Default is `false`.

`com.sun.identity.agents.config.trust.server.certs`

When SSL is configured, set to `false` to trust the OpenAM SSL certificate only if the certificate is found to be correct and valid. Default is `true` to make it easy to try SSL during evaluation.

Important

Notice that the default setting, `true`, means that the web policy agent trusts all server certificates. Change this to `false`, and test that your web policy agent can trust server certificates before deploying the policy agent in production.

`com.sun.identity.agents.config.username`

Set this to the user name of the agent authenticator.

`com.forgerock.agents.instance.id`

When there are multiple agents on a single system, set this to a unique numeric value.

1.2. Agent Configuration Properties

These properties are set in `config/agent.conf` if your agent uses local configuration. If your agent uses centralized configuration, the properties are set in OpenAM.

`com.forgerock.agents.agent.invalid.url.regex`

Set this property to a Perl-compatible regular expression to filter out invalid request URLs. The policy agent reject requests to invalid URLs with HTTP 403 Forbidden status without further processing.

For example, to filter out URLs containing the symbols in the list ./, /., /, ., ,\, %00-%1f, %7f-%ff, %25, %2B, %2C, %7E, .info, use the following setting.

```
com.forgerock.agents.agent.invalid.url.regex= \
 ^((?!(|/\.|\./||*|\.info|%25|%2B|%2C|%[0-1][0-9a-fA-F]|%[7-9a-fA-F][0-9a-fA-F])).)$
```

`com.forgerock.agents.agent.logout.url.regex`

Set this property to a Perl-compatible regular expression that matches logout URLs.

For example, to match URLs with `protectedA` or `protectedB` in the path and `op=logout` in the query string, use the following setting.

```
com.forgerock.agents.agent.logout.url.regex= \
  .*(/protectedA\?|/protectedB\?/).*(\&op=logout\&)(.*|$)
```

When you use this property, the agent ignores the settings for `com.sun.identity.agents.config.agent.logout.url`.

`com.forgerock.agents.cache_control_header.enable`

Set this property to `true` to enable use of Cache-Control headers that prevent proxies from caching resources accessed by unauthenticated users. Default: `false`.

`com.forgerock.agents.cdsso.disable.redirect.on_post`

Set this property to `true` to disable the HTTP 302 redirect after the LARES POST. By default, the policy agent does an HTTP redirect after processing the LARES POST message. Default: `false`.

This property applies only to Apache HTTPD 2.2 and 2.4 policy agents. Other policy agents do not redirect after processing the LARES POST message.

`com.forgerock.agents.conditional.login.url`

To conditionally redirect users based on the incoming request URL, set this property.

This takes the incoming request domain to match, a vertical bar (`|`), and then a comma-separated list of URLs to which to redirect incoming users.

If the domain before the vertical bar matches an incoming request URL, then the policy agent uses the list of URLs to determine how to redirect the user-agent. If the global property FQDN Check (`com.sun.identity.agents.config.fqdn.check.enable`) is enabled for the policy agent, then the policy agent iterates through the list until it finds an appropriate redirect URL that matches the FQDN check. Otherwise, the policy agent redirects the user-agent to the first URL in the list.

Property: `com.forgerock.agents.conditional.login.url`

Examples: `com.forgerock.agents.conditional.login.url[0]= login.example.com|http://openam1.example.com/openam/UI/Login, http://openam2.example.com/openam/UI/Login`, `com.forgerock.agents.conditional.login.url[1]= signin.example.com|http://openam3.example.com/openam/UI/Login, http://openam4.example.com/openam/UI/Login`

If CDSSO is enabled for the policy agent, then this property takes CDSSO Servlet URLs for its values (`com.sun.identity.agents.config.cdsso.cdcservlet.url`), rather than OpenAM login URLs.

CDSSO examples: `com.forgerock.agents.conditional.login.url[0]= login.example.com|http://openam1.example.com/openam/cdcservlet, http://openam2.example.com/openam/cdcservlet`, `com.forgerock.agents.conditional.login.url[1]= signin.example.com|http://openam3.example.com/openam/cdcservlet, http://openam4.example.com/openam/cdcservlet`

com.forgerock.agents.config.cert.ca.file

Set this property to the file name that contains one or more CA certificates. If `trust.server.certs=false`, the file should be PEM encoded.

> **Note**
>
> For OpenSSL, PEM format is base 64 encoded ASCII data. The acronym stands for Privacy Enhanced Mail format, as it was originally designed to secure email using public-key cryptography.

com.forgerock.agents.config.cert.file

Set this property to the name of the file that contains the PEM encoded public key certificate.

`com.forgerock.agents.config.cert.key`

Set this property to the name of the file that contains the private key. On UNIX systems, that key should be encoded in PEM format.

On Windows systems, that entry depends. If SSL mutual authentication is required with OpenAM, that entry should contain the name of the private key or certificate imported in the Windows Certificate Manager, part of the Microsoft Management Console. For a web server, that should point to the Local Machine or Service certificate store, depending on the account associated with the Web server.

com.forgerock.agents.config.ciphers

Set this property to the name of the cipher list. That list consists of one or more `cipher strings` separated by colons, as defined in the man page for `ciphers` available at http://www.openssl.org/docs/apps/ciphers.html.

Default: `HIGH:MEDIUM`.

Cipher restrictions can be configured as described in the Microsoft article entitled *How to Restrict the Use of Certain Cryptographic Algorithms and Protocols in Schannel.dll*.

`com.forgerock.agents.config.logout.redirect.disable`

Set this property to `true` to prevent the policy agent from redirecting to the logout URL when that logout URL matches one of the logout URL settings. Instead of redirecting the user-agent, the policy agent performs session logout in the background and then continues processing access to the current URL.

`com.forgerock.agents.config.notenforced.ip.handler`

As of version 3.0.4, web policy agents with this property set to `cidr` can use IPv4 netmasks and IP ranges instead of wildcards as values for `com.sun.identity.agents.config.notenforced.ip` addresses. Version 3.0.5 adds support for IPv6, including the IPv6 loopback address, `::1`.

When the parameter is defined, wildcards are ignored in `com.sun.identity.agents.config.notenforced.ip` settings. Instead, you can use settings such as those shown in the following examples.

Netmask Example

To disable policy agent enforcement for addresses in 192.168.1.1 to 192.168.1.255, use the following setting.

```
com.sun.identity.agents.config.notenforced.ip = 192.168.1.1/24
```

The following example shows a configuration using IPv6.

```
com.sun.identity.agents.config.notenforced.ip = 2001:5c0:9168:0:0:0:0:2/128
```

Currently the policy agent stops evaluating properties after reaching an invalid netmask in the list.

IP Range Example

To disable policy agent enforcement for addresses between 192.168.1.1 to 192.168.4.3 inclusive, use the following setting.

```
com.sun.identity.agents.config.notenforced.ip = 192.168.1.1-192.168.4.3
```

The following example shows a configuration using IPv6.

```
com.sun.identity.agents.config.notenforced.ip = 2001:5c0:9168:0:0:0:0:1-2001:5c0:9168:0:0:0:0:2
```

`com.forgerock.agents.config.pdpuri.prefix`

If you run multiple web servers with policy agents behind a load balancer that directs traffic based on the request URI, and you need to preserve POST data, then set this property.

By default, policy agents use a dummy URL for POST data preservation, `http://agent.host:port/dummypost/sunpostpreserve`, to handle POST data across redirects to and from OpenAM. When you set this property, the policy agent prefixes the property value to the dummy URL path. In other words, when you set `com.forgerock.agents.config.pdpuri.prefix = app1`, the policy agent uses the dummy URL, `http://agent.host:port/app1/dummypost/sunpostpreserve`.

Next, use the prefix you set when you define load balancer URI rules. This ensures that clients end up being redirected to the policy agent that preserved the POST data.

`com.forgerock.agents.notenforced.url.regex.enable`

Set this property to `true` to enable use of regular expressions in Not Enforced URL settings.

Not Enforced URL settings are configured using the property, `com.sun.identity.agents.config.notenforced.url`

Default: `false`

`com.sun.identity.agents.config.access.denied.url`

The URL of the customized access denied page. If no value is specified (default), then the agent returns an HTTP status of 403 (Forbidden).

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Resources Access Denied URL.

`com.sun.identity.agents.config.agent.logout.url`

List of application logout URLs, such as `http://www.example.com/logout.html`. The user is logged out of the OpenAM session when these URLs are accessed. When using this property, specify a value for the Logout Redirect URL property.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Logout URL List.

`com.sun.identity.agents.config.agenturi.prefix`

The default value is *`agent-root-URL`*`/amagent`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Agent Deployment URI Prefix.

`com.sun.identity.agents.config.anonymous.user.enable`

Enable or disable REMOTE_USER processing for anonymous users.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Anonymous User.

`com.sun.identity.agents.config.anonymous.user.id`

User ID of unauthenticated users. Default: `anonymous`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Anonymous User Default Value.

`com.sun.identity.agents.config.attribute.multi.value.separator`

Specifies separator for multiple values. Applies to all types of attributes such as profile, session and response attributes. Default: `|`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Attribute Multi Value Separator.

`com.sun.identity.agents.config.audit.accesstype`

Types of messages to log based on user URL access attempts.

Valid values for the configuration file property include `LOG_NONE`, `LOG_ALLOW`, `LOG_DENY`, and `LOG_BOTH`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Audit Access Types.

`com.sun.identity.agents.config.auth.connection.timeout`

Timeout period in seconds for an agent connection with OpenAM auth server. Default: 2

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Agent Connection Timeout.

`com.sun.identity.agents.config.cdsso.cdcservlet.url`

List of URLs of the available CDSSO controllers that the agent can use for CDSSO processing. For example, `http://openam.example.com:8080/openam/cdcservlet`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > CDSSO Servlet URL.

`com.sun.identity.agents.config.cdsso.cookie.domain`

List of domains, such as `.example.com`, in which cookies have to be set in CDSSO. If this property is left blank, then the fully qualified domain name of the cookie for the agent server is used to set the cookie domain, meaning that a host cookie rather than a domain cookie is set.

To set the list to `.example.com`, and `.example.net` using the configuration file property, include the following.

```
com.sun.identity.agents.config.cdsso.cookie.domain[0]=.example.com
com.sun.identity.agents.config.cdsso.cookie.domain[1]=.example.net
```

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cookies Domain List.

`com.sun.identity.agents.config.cdsso.enable`

Enables Cross Domain Single Sign On.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cross Domain SSO.

`com.sun.identity.agents.config.change.notification.enable`

Enables agent to receive notification messages from OpenAM server for configuration changes.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Profile.

`com.sun.identity.agents.config.cleanup.interval`

Interval in minutes to cleanup old agent configuration entries unless they are referenced by current requests. Default: 30.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Configuration Cleanup Interval.

`com.sun.identity.agents.config.client.hostname.header`

HTTP header name that holds the hostname of the client.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Client Hostname Header.

`com.sun.identity.agents.config.client.ip.header`

HTTP header name that holds the IP address of the client.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Client IP Address Header.

`com.sun.identity.agents.config.client.ip.validation.enable`

When enabled, validate that the subsequent browser requests come from the same IP address that the SSO token is initially issued against.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Client IP Validation.

`com.sun.identity.agents.config.convert.mbyte.enable`

When enabled, the agent encodes the LDAP header values in the default encoding of operating system locale. When disabled, the agent uses UTF-8.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Native Encoding of Profile Attributes.

`com.sun.identity.agents.config.cookie.name`

Name of the SSO Token cookie used between the OpenAM server and the agent. Default: `iPlanetDirectoryPro`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cookie Name.

`com.sun.identity.agents.config.cookie.reset.enable`

When enabled, agent resets cookies in the response before redirecting to authentication.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cookie Reset.

`com.sun.identity.agents.config.cookie.reset`

List of cookies in the format `name[=value][;Domain=value]`.

Concrete examples include the following with two list items configured.

- `LtpaToken`, corresponding to `com.sun.identity.agents.config.cookie.reset[0]=LtpaToken`. The default domain is taken from FQDN Default.
- `token=value;Domain=subdomain.domain.com`, corresponding to `com.sun.identity.agents.config.cookie.reset[1]= token=value;Domain=subdomain.domain.com`

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cookie Reset Name List.

`com.sun.identity.agents.config.cookie.secure`

When enabled, the agent marks cookies secure, sending them only if the communication channel is secure.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > SSO > Cookie Security.

`com.sun.identity.agents.config.debug.file.rotate`

When enabled, rotate the debug file when specified file size is reached.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Agent Debug File Rotation.

`com.sun.identity.agents.config.debug.file.size`

Debug file size in bytes beyond which the log file is rotated. The minimum is 1048576 bytes (1 MB), and lower values are reset to 1 MB. OpenAM console sets a default of 10 MB when it is used to configure the agent.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Agent Debug File Size.

`com.sun.identity.agents.config.debug.level`

Default is `Error`. Increase to `Message` or even `All` for fine-grained detail.

You can set the level in the configuration file by module using the format `module[:level][,module[:level]]*`, where *module* is one of `AuthService`, `NamingService`, `PolicyService`, `SessionService`, `PolicyEngine`, `ServiceEngine`, `Notification`, `PolicyAgent`, `RemoteLog`, or `all`, and *level* is one of the following.

- `0`: Disable logging from specified module

  At this level the agent nevertheless logs messages having the level value `always`.

- `1`: Log error messages
- `2`: Log warning and error messages
- `3`: Log info, warning, and error messages
- `4`: Log debug, info, warning, and error messages
- `5`: Like level 4, but with even more debugging messages

When you omit *level*, the agent uses the default level, which is the level associated with the `all` module.

The following example used in the local configuration sets the log overall level to debug for all messages.

```
com.sun.identity.agents.config.debug.level=all:4
```

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Agent Debug Level.

`com.sun.identity.agents.config.domino.check.name.database`

When enabled, the agent checks whether the user exists in the Domino name database.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Check User in Domino Database.

`com.sun.identity.agents.config.domino.ltpa.config.name`

The configuration name that the agent uses in order to employ the LTPA token mechanism.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > LTPA Token Configuration Name.

`com.sun.identity.agents.config.domino.ltpa.cookie.name`

The name of the cookie that contains the LTPA token.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > LTPA Token Cookie Name.

`com.sun.identity.agents.config.domino.ltpa.enable`

Enable if the agent needs to use LTPA Token.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Use LTPA token.

`com.sun.identity.agents.config.domino.ltpa.org.name`

The organization name to which the LTPA token belongs.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > LTPA Token Organization Name.

`com.sun.identity.agents.config.encode.cookie.special.chars.enable`

When enabled, encode special chars in cookie by URL encoding. This is useful when profile, session, and response attributes contain special characters, and the attributes fetch mode is set to `HTTP_COOKIE`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Encode special chars in Cookies.

`com.sun.identity.agents.config.encode.url.special.chars.enable`

When enabled, encodes the URL which has special characters before doing policy evaluation.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Encode URL's Special Characters.

`com.sun.identity.agents.config.fetch.from.root.resource`

When enabled, the agent caches the policy decision of the resource and all resources from the root of the resource down. For example, if the resource is `http://host/a/b/c`, then the root of the resource is `http://host/`. This setting can be useful when a client is expect to access multiple resources on the same path. Yet, caching can be expensive if very many policies are defined for the root resource.

Default: `false`

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Fetch Policies from Root Resource.

`com.sun.identity.agents.config.fqdn.check.enable`

Enables checking of FQDN default value and FQDN map values.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > FQDN Check.

`com.sun.identity.agents.config.fqdn.default`

Fully qualified domain name that the users should use in order to access resources. Without this value, the web server can fail to start, thus you set the property on agent installation, and only change it when absolutely necessary.

This property ensures that when users access protected resources on the web server without specifying the FQDN, the agent can redirect the users to URLs containing the correct FQDN.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > FQDN Default.

`com.sun.identity.agents.config.fqdn.mapping`

Enables virtual hosts, partial hostname and IP address to access protected resources. Maps invalid or virtual name keys to valid FQDN values so the agent can properly redirect users and the agents receive cookies belonging to the domain.

To map `myserver` to `myserver.mydomain.example`, enter `myserver` in the Map Key field, and enter `myserver.mydomain.example` in the Corresponding Map Value field. This corresponds to `com.sun.identity.agents.config.fqdn.mapping[myserver]= myserver.mydomain.example`.

Invalid FQDN values can cause the web server to become unusable or render resources inaccessible.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > FQDN Virtual Host Map.

`com.sun.identity.agents.config.get.client.host.name`

When enabled, get the client hostname through DNS reverse lookup for use in policy evaluation. This setting can impact performance.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Retrieve Client Hostname.

`com.sun.identity.agents.config.ignore.path.info`

When enabled, strip path info from the request URL while doing the Not Enforced List check, and URL policy evaluation. This is designed to prevent a user from accessing a URI by appending the matching pattern in the policy or not enforced list.

Note

This property is not supported by the Varnish Cache agent.

For example, if the not enforced list includes `http://host/*.gif`, then stripping path info from the request URI prevents access to `http://host/index.html` by using `http://host/index.html?hack.gif`.

However, when a web server is configured as a reverse proxy for a J2EE application server, the path info is interpreted to map a resource on the proxy server rather than the application server. This prevents the not enforced list or the policy from being applied to the part of the URI below the application server path if a wildcard character is used.

For example, if the not enforced list includes `http://host/webapp/servcontext/*` and the request URL is `http://host/webapp/servcontext/example.jsp`, the path info is `/servcontext/example.jsp` and the resulting request URL with path info stripped is `http://host/webapp/`, which does not match the not enforced list. Thus when this property is enabled, path info is not stripped from teh request URL even if there is a wildcard in the not enforced list or policy.

Make sure therefore when this property is enabled that there is nothing following the wildcard in the not enforced list or policy.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Ignore Path Info in Request URL.

`com.sun.identity.agents.config.ignore.path.info.for.not.enforced.list`

When enabled, the path info and query are stripped from the request URL before being compared with the URLs of the not enforced list for those URLs containing a wildcard character. This prevents a user from accessing `http://host/index.html` by requesting `http://host/index.html/hack.gif` when the not enforced list includes `http://host/*.gif`.

> **Note**
>
> This property is not supported by the Varnish Cache agent.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Ignore Path Info for Not Enforced URLs.

`com.sun.identity.agents.config.ignore.preferred.naming.url`

When enabled, do not send a preferred naming URL in the naming request.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Ignore Preferred Naming URL in Naming Request.

`com.sun.identity.agents.config.ignore.server.check`

When enabled, do not check whether OpenAM is up before doing a 302 redirect.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Ignore Server Check.

`com.sun.identity.agents.config.iis.auth.type`

The agent should normally perform authentication, so this is not required. If necessary, set to `none`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Authentication Type.

`com.sun.identity.agents.config.iis.filter.priority`

The loading priority of filter, DEFAULT, HIGH, LOW, or MEDIUM.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Filter Priority.

`com.sun.identity.agents.config.iis.owa.enable`

Enable if the IIS agent filter is configured for OWA.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Filter configured with OWA.

`com.sun.identity.agents.config.iis.owa.enable.change.protocol`

Enable to avoid IE6 security pop-ups.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Change URL Protocol to https.

`com.sun.identity.agents.config.iis.owa.enable.session.timeout.url`

URL of the local idle session timeout page.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Idle Session Timeout Page URL.

`com.sun.identity.agents.config.load.balancer.enable`

Enable if a load balancer is used for OpenAM services.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Load Balancer Setup.

`com.sun.identity.agents.config.local.log.rotate`

When enabled, audit log files are rotated when reaching the specified size.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Rotate Local Audit Log.

`com.sun.identity.agents.config.local.log.size`

Beyond this size limit in bytes the agent rotates the local audit log file if rotation is enabled. Default: 50 MB

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Local Audit Log Rotation Size.

`com.sun.identity.agents.config.locale`

The default locale for the agent.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Agent Locale.

`com.sun.identity.agents.config.log.disposition`

Specifies where audit messages are logged. By default, audit messages are logged remotely.

Valid values for the configuration file property include `REMOTE`, `LOCAL`, and `ALL`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Audit Log Location.

`com.sun.identity.agents.config.login.url`

OpenAM login page URL, such as `http://openam.example.com:8080/openam/UI/Login`, to which the agent redirects incoming users without sufficient credentials so then can authenticate.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > OpenAM Login URL.

`com.sun.identity.agents.config.logout.cookie.reset`

Cookies to be reset upon logout in the same format as the cookie reset list.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Logout Cookies List for Reset.

`com.sun.identity.agents.config.logout.redirect.url`

User gets redirected to this URL after logout. Specify this property alongside a Logout URL List.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Logout Redirect URL.

`com.sun.identity.agents.config.logout.url`

OpenAM logout page URL, such as `http://openam.example.com:8080/openam/UI/Logout`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > OpenAM Logout URL.

`com.sun.identity.agents.config.notenforced.ip`

No authentication and authorization are required for the requests coming from these client IP addresses.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Not Enforced Client IP List.

`com.sun.identity.agents.config.notenforced.url.attributes.enable`

When set to `true`, the agent fetches profile, response, and session attributes that are mapped by doing policy evaluation, and forwards these attributes to non-protected URLs.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Fetch Attributes for Not Enforced URLs.

`com.sun.identity.agents.config.notenforced.url.invert`

Only enforce not enforced list of URLs. In other words, enforce policy only for those URLs and patterns specified in the list.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Invert Not Enforced URLs.

`com.sun.identity.agents.config.notenforced.url`

List of URLs for which no authentication is required. You can use wildcards to define a pattern for a URL.

The `*` wildcard matches all characters except question mark (`?`), cannot be escaped, and spans multiple levels in a URL. Multiple forward slashes do not match a single forward slash, so `*` matches `mult/iple/dirs`, yet `mult/*/dirs` does not match `mult/dirs`.

The `-*-` wildcard matches all characters except forward slash (`/`) or question mark (`?`), and cannot be escaped. As it does not match `/`, `-*-` does not span multiple levels in a URL.

OpenAM does not let you mix `*` and `-*-` in the same URL.

Examples include `http://www.example.com/logout.html`, `http://www.example.com/images/*`, `http://www.example.com/css/-*-`, and `http://www.example.com/*.jsp?locale=*`.

Trailing forward slashes are not recognized as part of a resource name. Therefore `http://www.example.com/images//` and `http://www.example.com/images` are equivalent.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Not Enforced URLs.

If you set `com.forgerock.agents.notenforced.url.regex.enable=true`, then you can use Perl-compatible regular expressions to match Not Enforced URLs instead. (Do not mix settings; use either the mechanism described above or Perl-compatible regular expressions, but not both.)

The following example shows settings where no authentication is required for URLs whose path ends `/publicA` or `/publicB` (with or without query string parameters), and no authentication is required to access .png, .jpg, .gif, .js, or .css files under URLs that do not contain `/protectedA/` or `/protectedB/`.

```
com.forgerock.agents.notenforced.url.regex.enable=true
com.sun.identity.agents.config.notenforced.url[0]= \
 .*/(PublicServletA|PublicServletB)(\?.*|$)
com.sun.identity.agents.config.notenforced.url[1]= \
 ^(?!.*(/protectedA/|/protectedB/)).*\.(png|jpg|gif|js|css)(\?.*|$)
```

com.sun.identity.agents.config.notification.enable

If enabled, the agent receives policy updates from the OpenAM notification mechanism to maintain its internal cache. If disabled, the agent must poll OpenAM for changes.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Enable Notifications.

com.sun.identity.agents.config.override.host

Enable if the agent is sitting behind a SSL/TLS off-loader, load balancer, or proxy such that the host name users use is different from the host name the agent uses. When enabled, the host is overridden with the value from the Agent Deployment URI Prefix (property: `com.sun.identity.agents.config.agenturi.prefix`).

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Override Request URL Host.

com.sun.identity.agents.config.override.notification.url

Enable if the agent is sitting behind a SSL/TLS off-loader, load balancer, or proxy such that the URL users use is different from the URL the agent uses. When enabled, the URL is overridden with the value from the Agent Deployment URI Prefix (property: `com.sun.identity.agents.config.agenturi.prefix`).

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Override Notification URL.

com.sun.identity.agents.config.override.port

Enable if the agent is sitting behind a SSL/TLS off-loader, load balancer, or proxy such that the port users use is different from the port the agent uses. When enabled, the port is overridden with the value from the Agent Deployment URI Prefix (property: `com.sun.identity.agents.config.agenturi.prefix`).

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Override Request URL Port.

com.sun.identity.agents.config.override.protocol

Enable if the agent is sitting behind a SSL/TLS off-loader, load balancer, or proxy such that the protocol users use is different from the protocol the agent uses. When enabled, the protocol is overridden with the value from the Agent Deployment URI Prefix (property: `com.sun.identity.agents.config.agenturi.prefix`).

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Override Request URL Protocol.

`com.sun.identity.agents.config.policy.cache.polling.interval`

Polling interval in minutes during which an entry remains valid after being added to the agent's cache.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Policy Cache Polling Period.

`com.sun.identity.agents.config.policy.clock.skew`

Time in seconds used adjust time difference between agent system and OpenAM. Clock skew in seconds = AgentTime - OpenAMServerTime.

Use this property to adjust for small time differences encountered despite use of a time synchronization service. When this property is not set and agent time is greater than OpenAM server time, the agent can make policy calls to the OpenAM server before the policy subject cache has expired, or you can see infinite redirection occur.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Policy Clock Skew.

`com.sun.identity.agents.config.poll.primary.server`

Interval in minutes, agent polls to check the primary server is up and running. Default: 5.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > Polling Period for Primary Server.

`com.sun.identity.agents.config.polling.interval`

Interval in minutes to fetch agent configuration from OpenAM. Used if notifications are disabled. Default: 60.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Configuration Reload Interval.

`com.sun.identity.agents.config.postcache.entry.lifetime`

POST cache entry lifetime in minutes. Default: 10.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > POST Data Entries Cache Period.

`com.sun.identity.agents.config.postdata.preserve.enable`

Enables HTTP POST data preservation. This feature is available in the Apache 2.2, Microsoft IIS 6, Microsoft IIS 7, and Sun Java System Web Server web policy agents as of version 3.0.3.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > POST Data Preservation.

`com.sun.identity.agents.config.postdata.preserve.lbcookie`

When HTTP POST data preservation is enabled, override properties are set to true, and the agent is behind a load balancer, then this property sets the name and value of the sticky cookie to use.

`com.sun.identity.agents.config.postdata.preserve.stickysession.mode`

Specifies whether to create a cookie, or to append a query string to the URL to assist with sticky load balancing.

`com.sun.identity.agents.config.postdata.preserve.stickysession.value`

Specifies the key-value pair for stickysession mode. For example, a setting of `lb=myserver` either sets an `lb` cookie with `myserver` value, or adds `lb=myserver` to the URL query string.

`com.sun.identity.agents.config.profile.attribute.cookie.maxage`

Maximum age in seconds of custom cookie headers. Default: 300.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Profile Attributes Cookie Maxage.

`com.sun.identity.agents.config.profile.attribute.cookie.prefix`

Sets cookie prefix in the attributes headers. Default: `HTTP_`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Profile Attributes Cookie Maxage.

`com.sun.identity.agents.config.profile.attribute.fetch.mode`

When set to `HTTP_COOKIE` or `HTTP_HEADER`, profile attributes are introduced into the cookie or the headers, respectively.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Profile Attribute Fetch Mode.

`com.sun.identity.agents.config.profile.attribute.mapping`

Maps the profile attributes to HTTP headers for the currently authenticated user. Map Keys are LDAP attribute names, and Map Values are HTTP header names.

To populate the value of profile attribute CN under `CUSTOM-Common-Name`: enter CN in the Map Key field, and enter `CUSTOM-Common-Name` in the Corresponding Map Value field. This corresponds to `com.sun.identity.agents.config.profile.attribute.mapping[cn]=CUSTOM-Common-Name`.

In most cases, in a destination application where an HTTP header name shows up as a request header, it is prefixed by `HTTP_`, lower case letters become upper case, and hyphens (`-`) become underscores (`_`). For example, `common-name` becomes `HTTP_COMMON_NAME`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Profile Attribute Map.

`com.sun.identity.agents.config.proxy.override.host.port`

When enabled ignore the host and port settings for Sun Java System Proxy.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Override Proxy Server's Host and Port.

`com.sun.identity.agents.config.redirect.param`

Property used only when CDSSO is enabled. Only change the default value, `goto` when the login URL has a landing page specified such as, `com.sun.identity.agents.config.cdsso.cdcservlet.url = http://openam.example.com:8080/openam/cdcservlet?goto= http://www.example.com/landing.jsp`. The agent uses this parameter to append the original request URL to this cdcservlet URL. The landing page consumes this parameter to redirect to the original URL.

As an example, if you set this value to `goto2`, then the complete URL sent for authentication is `http://openam.example.com:8080/openam/cdcservlet?goto= http://www.example.com/landing.jsp?goto2=http://www.example.com/original.jsp`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > Goto Parameter Name.

`com.sun.identity.agents.config.remote.log.interval`

Periodic interval in minutes in which audit log messages are sent to the remote log file. Default: 5

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Remote Audit Log Interval.

`com.sun.identity.agents.config.remote.logfile`

Name of file stored on OpenAM server that contains agent audit messages if log location is remote or all.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Remote Log Filename.

`com.sun.identity.agents.config.replaypasswd.key`

DES key for decrypting the basic authentication password in the session for IIS.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Advanced > Replay Password Key.

`com.sun.identity.agents.config.repository.location`

Whether the agent's configuration is managed centrally through OpenAM (`centralized`) or locally in the policy agent configuration file (`local`).

Default: `centralized`

`com.sun.identity.agents.config.response.attribute.fetch.mode`

When set to `HTTP_COOKIE` or `HTTP_HEADER`, response attributes are introduced into the cookie or the headers, respectively.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Response Attribute Fetch Mode.

`com.sun.identity.agents.config.response.attribute.mapping`

Maps the policy response attributes to HTTP headers for the currently authenticated user. The response attribute is the attribute in the policy response to be fetched.

To populate the value of response attribute `uid` under `CUSTOM-User-Name`: enter `uid` in the Map Key field, and enter `CUSTOM-User-Name` in the Corresponding Map Value field. This corresponds to `com.sun.identity.agents.config.response.attribute.mapping[uid]=Custom-User-Name`.

In most cases, in a destination application where an HTTP header name shows up as a request header, it is prefixed by `HTTP_`, lower case letters become upper case, and hyphens (`-`) become underscores (`_`). For example, `response-attr-one` becomes `HTTP_RESPONSE_ATTR_ONE`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Response Attribute Map.

`com.sun.identity.agents.config.session.attribute.fetch.mode`

When set to `HTTP_COOKIE` or `HTTP_HEADER`, session attributes are introduced into the cookie or the headers, respectively.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Session Attribute Fetch Mode.

`com.sun.identity.agents.config.session.attribute.mapping`

Maps session attributes to HTTP headers for the currently authenticated user. The session attribute is the attribute in the session to be fetched.

To populate the value of session attribute `UserToken` under `CUSTOM-userid`: enter `UserToken` in the Map Key field, and enter `CUSTOM-userid` in the Corresponding Map Value field. This corresponds to `com.sun.identity.agents.config.session.attribute.mapping[UserToken] =CUSTOM-userid`.

In most cases, in a destination application where an HTTP header name shows up as a request header, it is prefixed by `HTTP_`, lower case letters become upper case, and hyphens (`-`) become underscores (`_`). For example, `success-url` becomes `HTTP_SUCCESS_URL`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Application > Session Attribute Map.

`com.sun.identity.agents.config.sso.cache.polling.interval`

Polling interval in minutes during which an SSO entry remains valid after being added to the agent's cache.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > SSO Cache Polling Period.

`com.sun.identity.agents.config.sso.only`

When enabled, agent only enforces authentication (SSO), but no policies for authorization.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > SSO Only Mode.

`com.sun.identity.agents.config.url.comparison.case.ignore`

When enabled, enforces case insensitivity in both policy and not enforced URL evaluation.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Miscellaneous > URL Comparison Case Sensitivity Check.

`com.sun.identity.agents.config.userid.param`

Agent sets this value for User Id passed in the session from OpenAM to the REMOTE_USER server variable. Default: UserToken.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > User ID Parameter.

`com.sun.identity.agents.config.userid.param.type`

User ID can be fetched from either SESSION and LDAP attributes. Default: `SESSION`.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > OpenAM Services > User ID Parameter Type.

`com.sun.identity.client.notification.url`

URL used by agent to register notification listeners.

For centralized configurations this property is configured under Access Control > *Realm Name* > Agents > Web > *Agent Name* > Global > Agent Notification URL.

`com.sun.identity.cookie.httponly`

Set this property to `true` to mark `iPlanetDirectoryPro` cookies as HTTPOnly, preventing scripts and third-party programs from accessing the cookies.