CAS1 Plugin

Plugin Information
Distribution of this plugin has been suspended due to unresolved security vulnerabilities, see below.

The current version of this plugin may not be safe to use. Please review the following warnings before use:

- Arbitrary code execution vulnerability

General

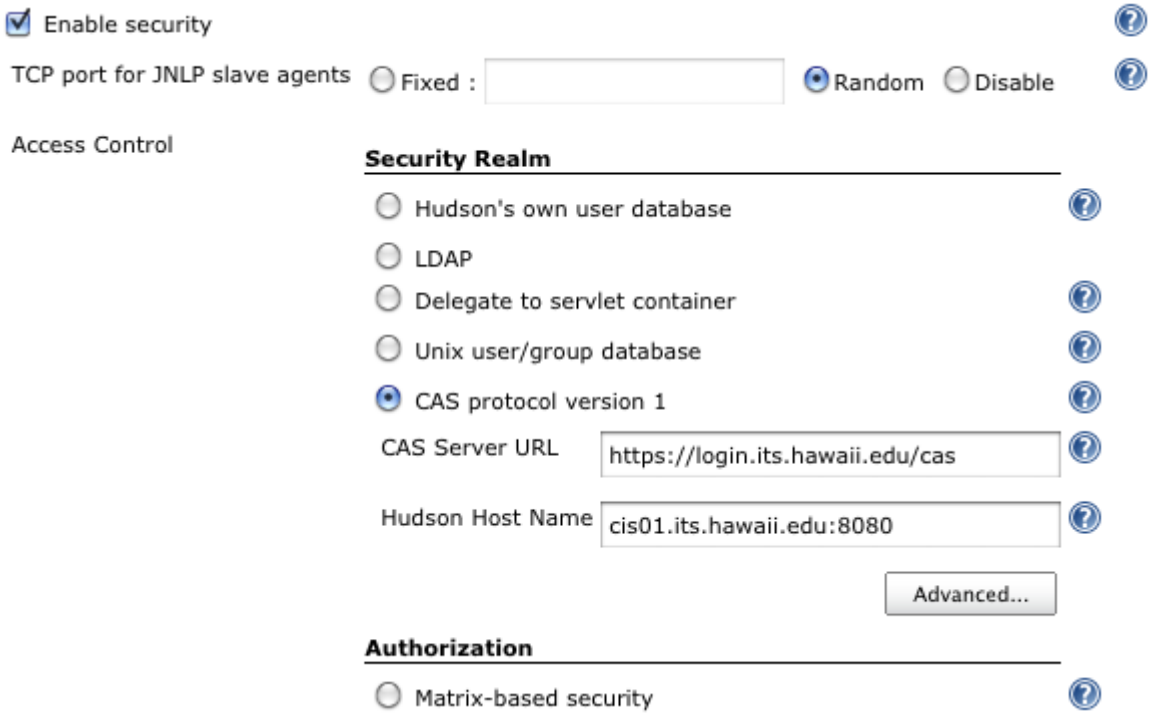
This plugin is obsolete; use the CAS Plugin instead.
This plugin supports only the oldest version of the CAS protocol, and may be incompatible with newer versions and features of Jenkins. It has not been maintained for many years. It was superseded by the CAS Plugin, which should support all of its features on the CAS 1 protocol, as well as newer CAS protocols and Jenkins versions. (These plugins have separate configurations, so if you upgrade to the CAS Plugin, you will need to configure it too.) The documentation below has not been updated.

This plugin lets Jenkins authenticate users via your organization's Central Authentication Service (CAS), for single-sign-on. It adds a Security Realm for the CAS protocol version 1 (plain text), which should be compatible with all versions of CAS. It also allows you to configure a Groovy script that determines a user's authorities/roles/groups. This script could work by parsing custom extensions in your CAS validation response, such as LDAP affiliation details.

Setup

Basic Setup

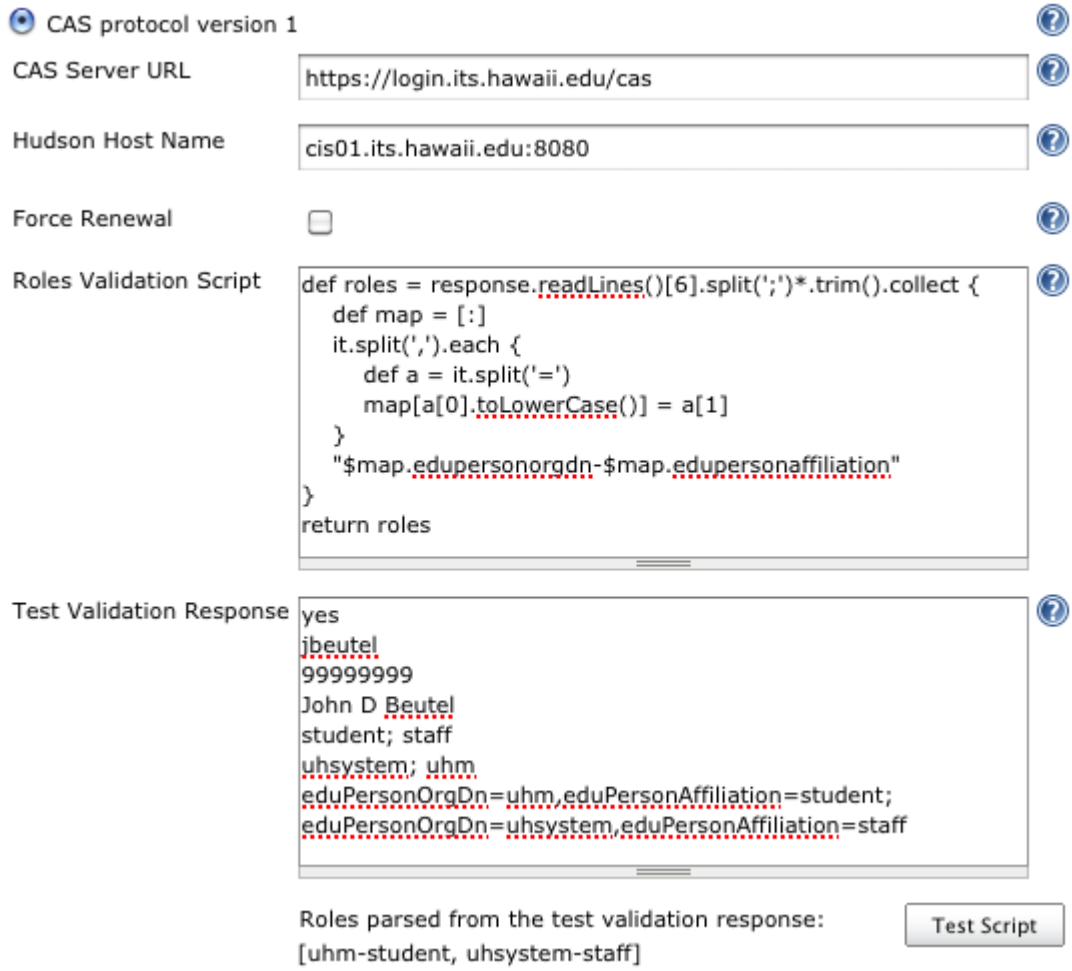
1. if your CAS restricts the services for which it provides authentication, register your Jenkins service URL with your CAS
2. Manage Jenkins > Manage Plugins > Available > install CAS1 plugin
3. Manage Jenkins > Configure System > Enable security
4. select the CAS protocol version 1 Security Realm
5. input the URL of your CAS server and the host name/port number of your Jenkins server
6. click focus on another field so AJAX will validate your input
7. heed warnings on your input, if any
8. click the Save button at the bottom if there are no warnings



Advanced Setup

1. click the Advanced... button under CAS protocol version 1
2. input a Groovy script that determines the list of groups/roles of any given user
3. input an example validation response from your CAS
4. click the Test Script and confirm the list of groups/roles your script produced
5. select "Project-based Matrix Authorization Strategy" or "Matrix based security" and add groups matching roles returned by your script
6. be sure to give yourself the Administer permission
7. click the Save button at the bottom if there are no warnings

The example below is for a custom CAS server validation response, containing extra details from LDAP, including affiliation. (The last two lines of the Test Validation Response is actually a single line displayed as wrapped by the narrow browser window.) For cut-and-paste, this example is also in the help text (? icon).



Another example script determines roles from a standard validation response and ad hoc lists of users. It can also be combined with the above example script.

```
def username = response.readLines()[1].trim()
roles += [
    'hudson-adm': ['jbeutel', 'jdoe', 'rsmith'],
    'developer': ['jbeutel', 'jdoe', 'sclaus', 'ebunny'],
    'tester': ['itokugawa', 'hmatsu'] // etc...
].collect { role, names -> names.contains(username) ? role : [] }.
flatten()
return roles
```

Limitations

- This Security Realm authenticates all pages; it has not implemented anonymous access. So, the distinction between the Authorization choices of "Logged-in users can do anything" and "Anyone can do anything" is lost; the latter becomes the former. Likewise, for the Authorization choices of "Matrix based security" and "Project-based Matrix Authorization Strategy", the mandatory "Anonymous" user/group is superfluous and redundant with the build-in "authenticated" role.
- It does not support CAS protocol version 2 (XML), including proxies or attributes. (It looks like a plugin for all that could be implemented with just the Acegi library that comes with Jenkins, but Acegi does not seem to support version 1 of the CAS protocol, so this plugin includes the Java CAS client library instead.)
- The plugin will initiate authentication on any page. If your CAS restricts which pages it is willing to authenticate, then your users may need to start on one of those pages of Jenkins.

Change Log

Version 1.0.1 (2010 Mar 9)

- testing Update Center

Version 1.0 (2010 Feb 26)

- initial release