

# CAS

- [CAS \(Central Authentication Service\)](#)
- [CAS architecture](#)
- [CAS Example](#)

# CAS (Central Authentication Service)

## Introduction

“ The CAS protocol is a simple and powerful ticket-based protocol. It involves one or many clients and one server. Clients are embedded in CASified applications (called “CAS services”) whereas the CAS server is a standalone component.

The Cas protocol makes it possible to implement the SSO authentication method that allows users to access web applications with a single sign-on.

The specification versions recognized are 3.0.3 and 2.0



---

<https://apereo.github.io/cas/6.5.x/protocol/CAS-Protocol.html>

# CAS architecture

## Introduction

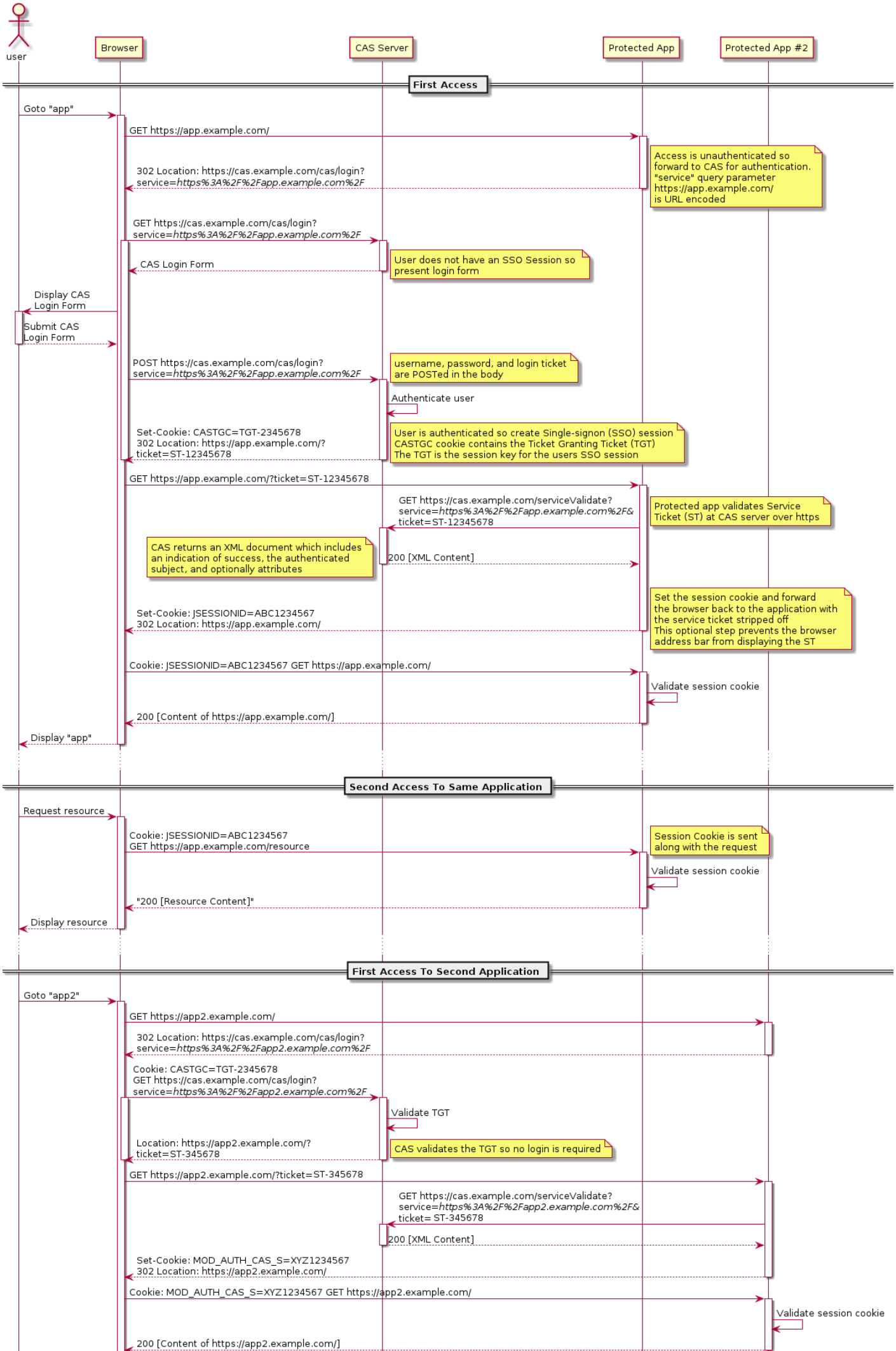
The CAS is a Single Sign On protocol for the web. This protocol allows users to access multiple applications by providing their credentials.

The response will be a JSON or XML

## Single Log-in

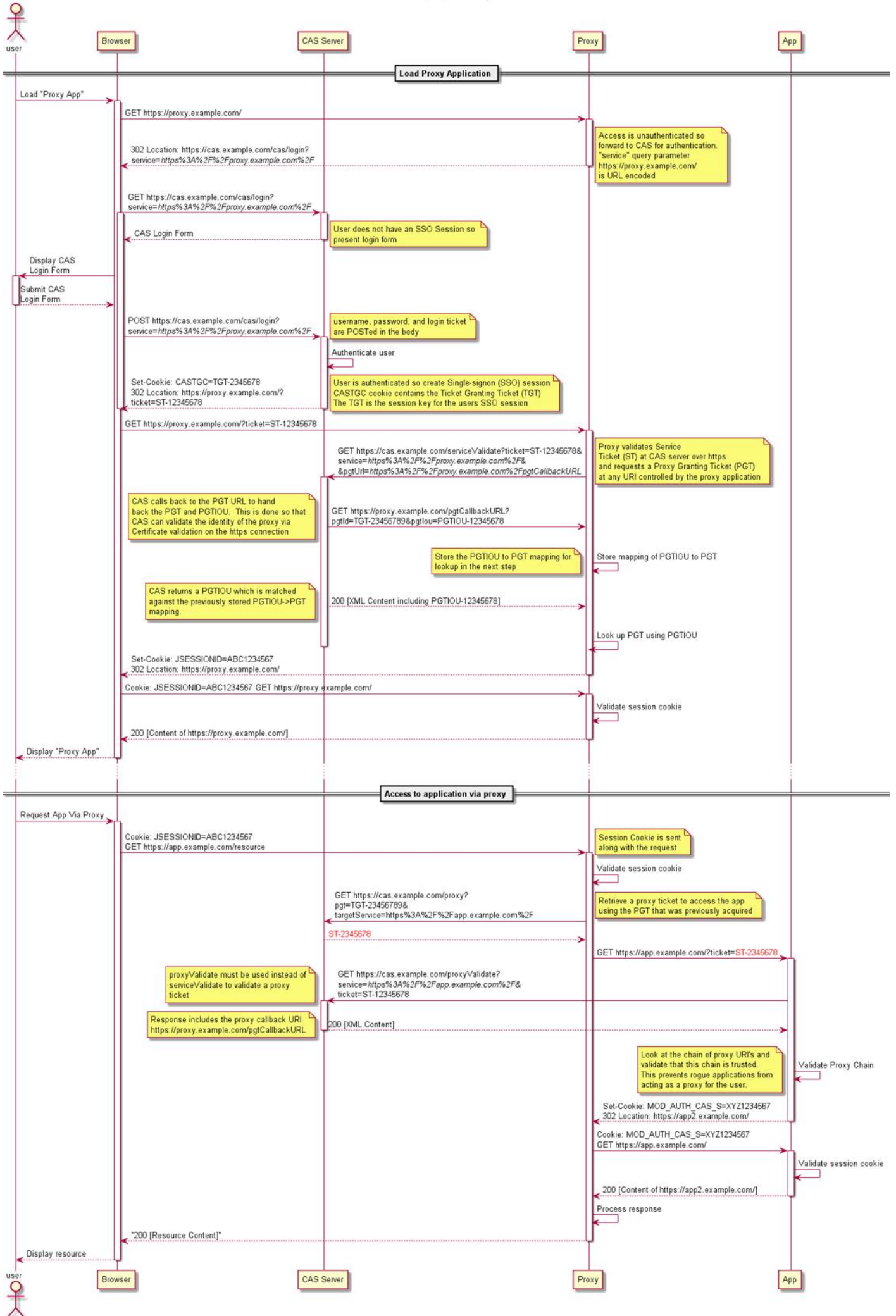
The single log-in is usually initiated by the application server. The typical UML use case is as follows:

# CAS Browser Single-Signon Sequence Diagram



# Proxy web flow diagram

# CAS Proxy Sequence Diagram



---

[https://en.wikipedia.org/wiki/Central\\_Authentication\\_Service](https://en.wikipedia.org/wiki/Central_Authentication_Service)

# CAS Example

## Service Provider

### Identification

Type :	CAS client
publicID :	http://127.0.1.1/
Name :	CAS client

### CAS Configuration

Response URL :	http://127.0.1.1/
	Response URL
Logout response URL :	Logout response URL

### Login rules

Allow impersonations :	Target application URL
UID Script :	Script to compute the user name to pass to the target application
Ask for consent :	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No
Roles required to login :	Roles required to login
System where an enabled account is required :	

Undo Apply changes