

# Authentication - Authorization

- [General](#)
  - [Token Passing](#)
  - [Signed URLs](#)
  - [Encrypted streams](#)
  - [Red5 authentication](#)
  - [Adobe FlashAccess](#)
  - [HydraHead authorization](#)
- [Embedded player auth](#)
  - [Flash + JS](#)
  - [IFrame](#)

[Cross-team discussion with PSDS/DIL \(NU\) and Avalon \(NU-IU\) teams, August 2, 2012](#)

## General

Story [VOV-434 - Getting issue details...](#)  indicates the needs for 3 types of group: public, authenticated and staff. There will be content viewable by anyone, or by authenticated users only, or by staff only.

## Token Passing

Avalon generates a token and attaches it to the video URL.

Streaming server calls back to Avalon to verify authenticity of token and validity for requested stream.

[Token/Callback Authentication Details](#)

## Signed URLs

Calculate an MD5 signature based on the required expiration time, the path of the link, and a secret key of your account.

Attach signature to video URL.

If signature in URL request matches signature calculated on server then proceed with the streaming.

## Encrypted streams

RTMPS over secure SSL connection. [Natively supported in Red5.](#)

RTMPE vulnerable to Man in the middle attacks. Programs like `rtmpdump` can still record `rtmpe` streams.

## Red5 authentication

See [Red5 authentication](#)

## Adobe FlashAccess

[Features](#) supported on FlashPlayer 10.1 and newer

## HydraHead authorization

There is a [Hydra ticket](#) for future implementation of [CanCan](#) - a great authorization lib for Rails.

Hydra ND Video Head seems to [have used it](#).

## Embedded player auth

What happens when someone wants to embed a video with restricted access on a 3rd party website (ie professorX.com)? We need to enable auth in embedded player.

## Flash + JS

Host an swf file on IU server, a script on professorX.com fetches the swf and uses it to read IU auth cookies.

If user has previously logged in, cookie is there, access should be granted. If not, open an IU pop-up, ask user to login.

## IFrame

Previous method may not run on iOS. Youtube has moved to use iframe to embed videos.

If we embed the player as an iframe, the iframe can read cookie from IU, with the exception of Safari, for which [a workaround exists](#)

## Cross-team discussion with PSDS/DIL (NU) and Avalon (NU-IU) teams, August 2, 2012

- Prior to the meeting:
  - Review key documents in the [DIL Working Documents Google docs folder](#): (contact Claire or Andrea for access)
    - List of Access Types used in Hydra-DIL, in the SECOND TAB (labeled 'Access Types & Scenarios') of the [Repo Policy Access Grid](#) document
    - Towards the end of our work with MediaShelf, we started talking about permitting more than one policy per repository object, and also about supporting Admin Collections. In order to understand what this means, it would be helpful to read at least the first few pages of [DIL Access Control Policies](#)
    - There are some UI wireframes in the folder that might be worth a quick glance
    - [Access Control Starter Stories](#) will give you a sense of how we tried to write interface descriptions for this work
  - Read Avalon authorization stories [VOV-434](#) and [VOV-435](#)
- Mike Stroming will talk about the work MediaShelf did in the PSDS/DIL sprint, and the general implications for authorization. He may not be able to demo via the testing server, but he can talk about the work and the middleware
- Group discussion: what questions are there about how to proceed with the Avalon work for this sprint, and for authn/authz in general?