ASWF Zero Trust WG

April 3, 2024

Jim Helman, Daryll Strauss, Chris Vienneau, Matt Daw, Spencer Stephens
MovieLabs

Background & Purpose

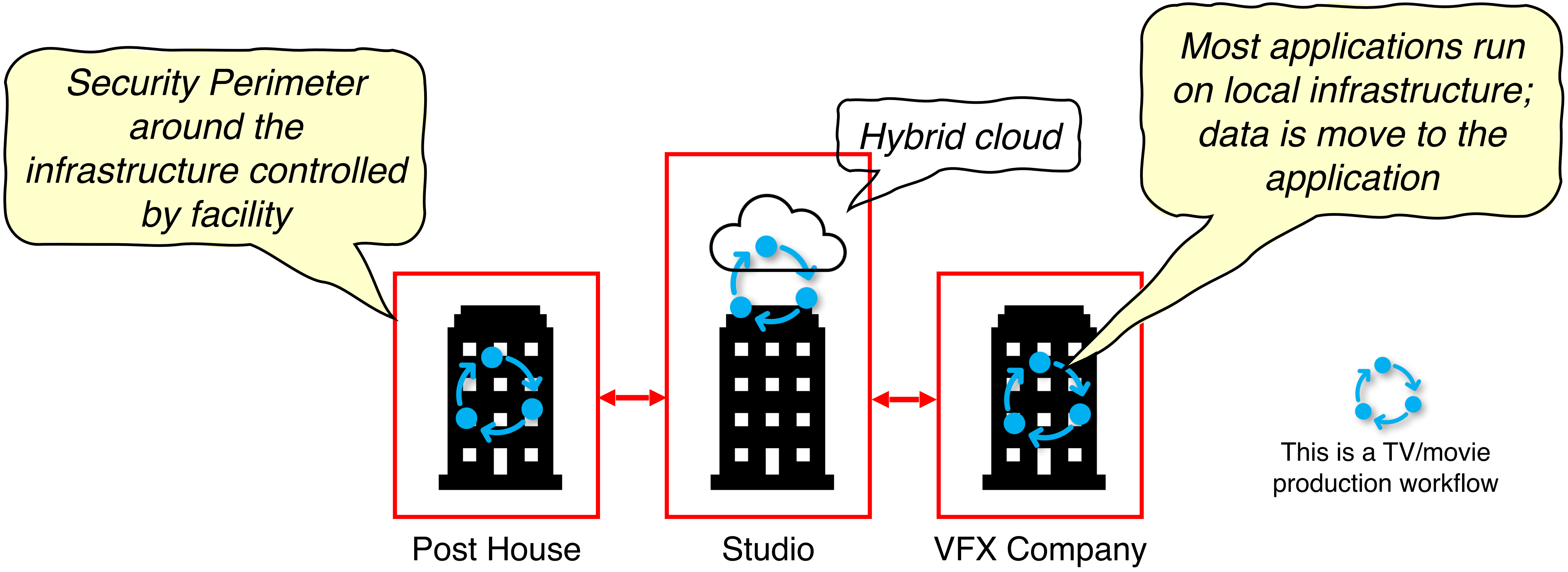
Legacy solutions relied on perimeter security for media creation: Secure the facility with firewalls and have artists work entirely inside the perimeter with minimal additional security checks.

As the complexity of productions has increased and attacks have evolved, the perimeter security model has become less effective. Zero Trust solutions, which add end to end security between the user and the service, improve both the efficiency and security of the production.

This working group will develop standards and best practices for implementing zero trust solutions in media production applications.

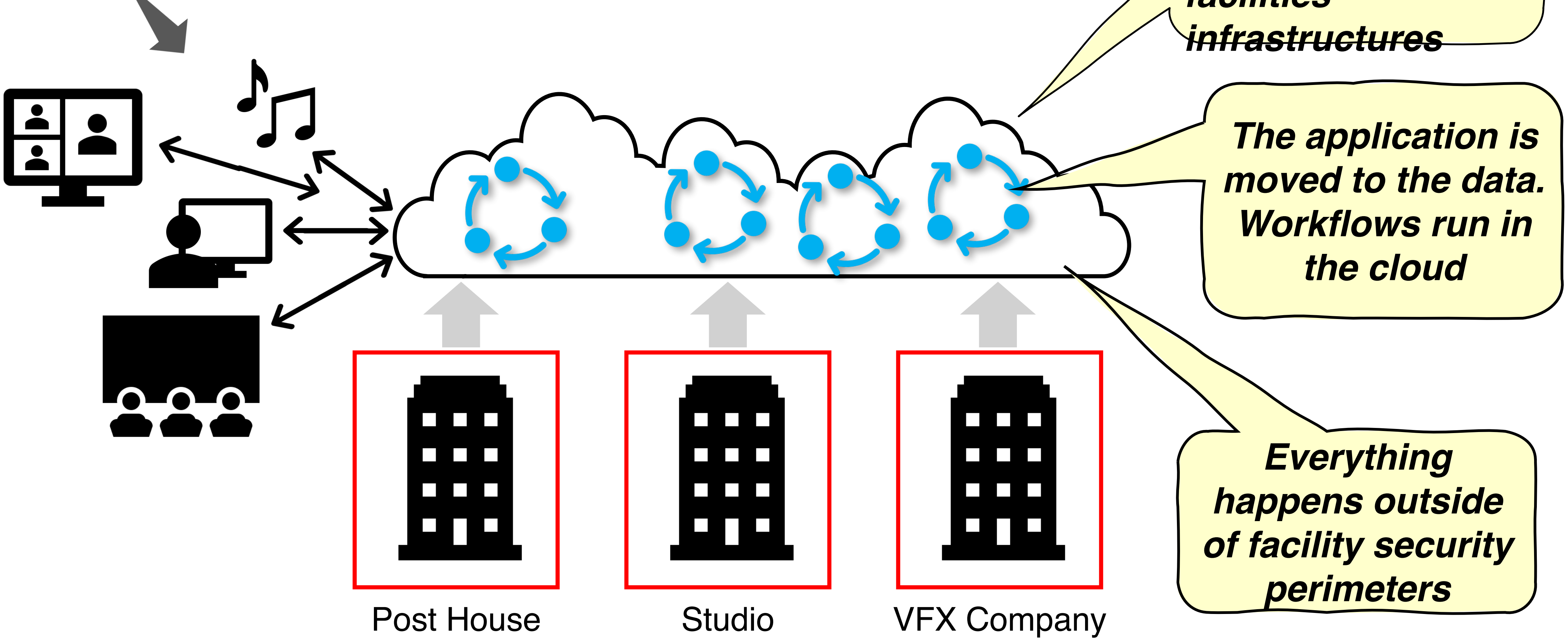
Proposal: https://github.com/AcademySoftwareFoundation/tac/issues/621

Today's production infrastructure: Datacenter and hybrid cloud


Security Perimeter around the infrastructure controlled by facility
Hybrid cloud
Most applications run on local infrastructure; data is move to the application
Post House
Studio
VFX Company
This is a TV/movie production workflow

Production ***in the*** cloud

Individual contributors & small vendors connected directly to the production cloud



The cloud is a global production resource outside of facilities' infrastructures

The application is moved to the data. Workflows run in the cloud

Everything happens outside of facility security perimeters

The Problem

- Multiple facilities, data centers, cloud computing, and SaaS applications make defining perimeters difficult.
- Passing data between vendors on the production makes the perimeters porous.
- BYOD, work from home, and working on location increase efficiency but are all challenges to perimeter security.
- VPNs require security management, lower performance, and provide little to no granularity of access.
- Applications and plugins increasingly rely on network services making defining and managing the perimeter more difficult.
- Automation and software-defined workflows improve efficiency but also increase the API connections across infrastructures and organizations.

The Solution: Zero Trust (ZT)

- In ZT, nothing is trusted without first being authenticated
- ZT authenticates a user, and perhaps a machine, rather than a just trusting everything inside a perimeter.
- In ZT, authentication is mutual. The service authenticates the user/client. The client authenticates the server.
    - Client applications and plug-ins using network services need to incorporate ZT, acquiring and transmitting authentication tokens, and authenticating the server (e.g., via HTTPS its certificate).
    - Network services incorporate ZT by authenticating the user/client and checking that the user is authorized to do the activity.
- Sharing authentication data between media applications, with a single sign on, reduces friction for the artist.

Standardization Benefits

- Make implementing ZT solutions easier.
- Avoid duplication of effort for software ASWF projects.
- Increase security by having trusted implementations and recommended policies.
- Lower the impact on artists by reducing the number of separate authentication requests from applications.

Engagement

- Participation
  - MovieLabs, Daryll Strauss, Chris Vienneau, *Spencer Stephens, Matt Daw*
  - Amazon, Blake Franzen
  - Google, Toby Scales
  - Foundry, Dan Hutchinson
  - Autodesk, Claude Robillard
  - Adobe, TBD
- Relevant ASWF Projects
  - Used with plug-ins accessing network services: OpenAssetIO
  - Client/server systems: OpenCue (c.f. issues 218, 344, 425)
  - For currently non-core use cases: OpenReview (already patched w/one approach), OpenImageIO (if I/O proxy used to access network resources)
  - Potential projects waiting in the wings: Google OpenProductionDataIO