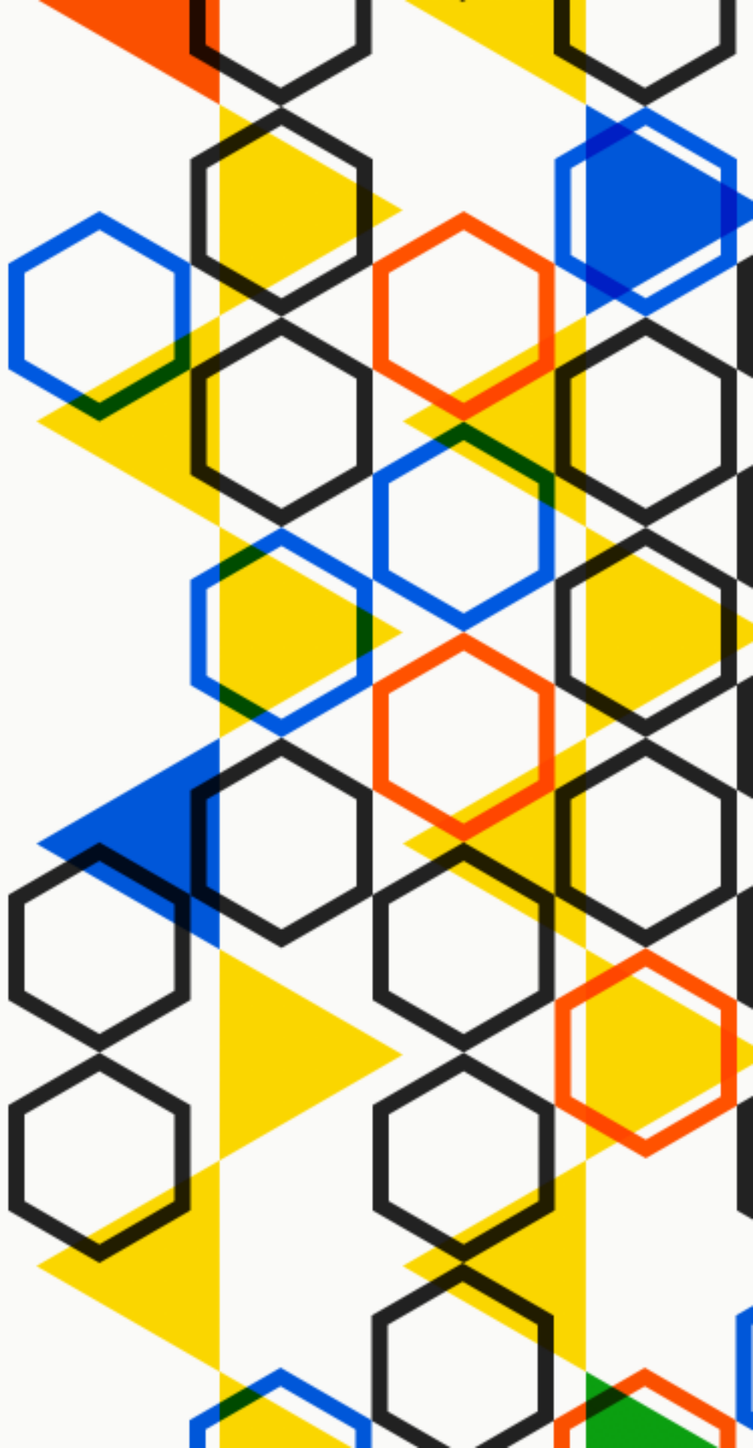
cloud.gov

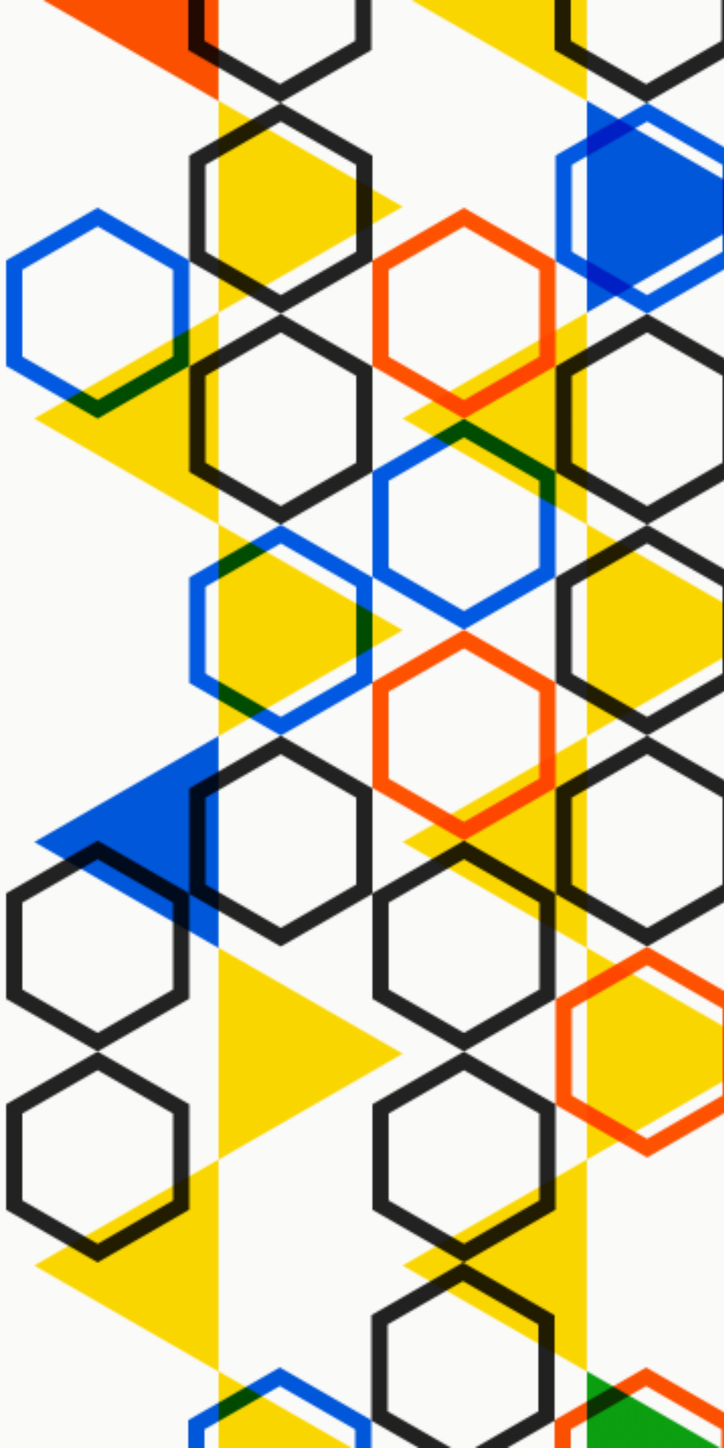
Hands-on workshop with cloud.gov



09:00	**Welcome** Shashank Khandelwal
09:10	**cloud.gov Overview**
09:40	**cloud.gov Hands-on I**
10:20	**Break**
10:30	**Federalist** Will Slack
10:40	**cloud.gov Hands-on II**
11:30	**Q & A**

github.com/18F/cg-workshop

CLOUD.GOV



I Want You

to use cloud.gov

— ➡️: Focus on mission

— 🕑: Eliminate long lead times

— 💵: Your tax 💰 ($85B, 8.2% ☁)[1]

— 🇺🇸: Provide great public service

[1] CIO IT Dashboard for FY2017 https://www.itdashboard.gov/#learn-basic-stats

1 / The Mission

Video timestamp 04:02

Suppose:

- **A mission**
  - Housing for disaster victims
- **A team**
  - Project / Product Managers
    Designers / Devs
    Ops / Sec
- **A platform**
  - **Build**
  - **Test**
  - **Run**

Video timestamp 04:19

Platform

— **Stack:** WebServer, AppServer, Database, Cache, Index

— **Environments:** (Local), Dev, Test, Stage, Prod

— **User** management: Admin, Devs, Auditors

— **Operations:** Patch, Logs, CDN, Scaling, Availability

— **All of this is commodity**: think iPad or Android Tablet

— **Acquire: weeks // Running: hours // Build: months // Authorize: weeks**

Video timestamp 04:52

CLOUD.GOV

- **Open-source Cloud Foundry PaaS atop AWS GovCloud[2]**
- **Available to Departments & Agencies by IAA**
- **FedRAMP P-ATO Moderate, DISA Level 2**
- **Built/run by 18F/TTS/GSA as a cost-recoverable service**

[2] Multi-cloud w/ Azure USGov on our roadmap

Video timestamp 07:15

Pre-built environment ready for deploying an application.

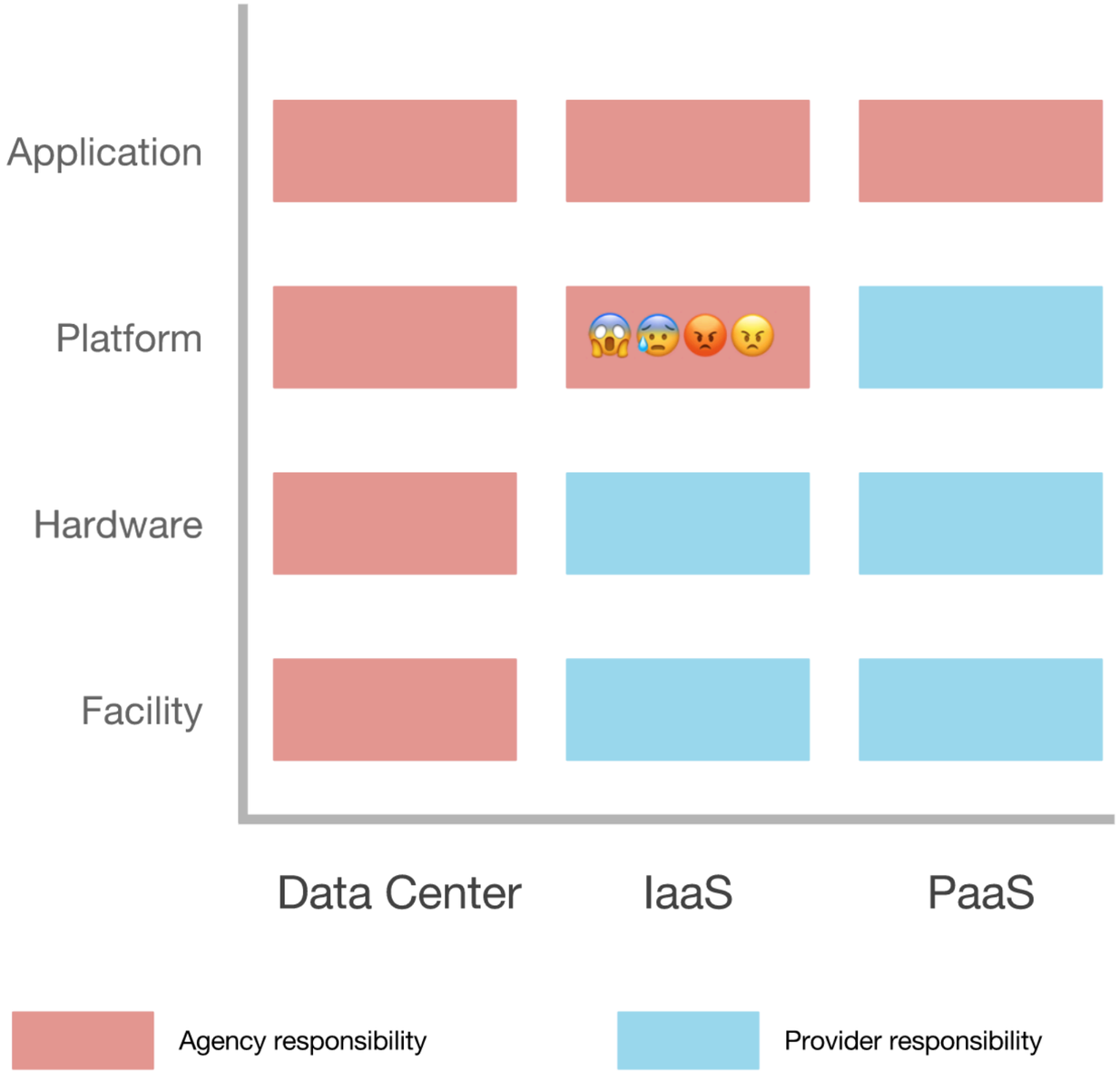
Developers can focus on mission needs.

Common technology resources are managed by an expert operations team:

— Operating system
— Databases
— Audit trails
— Authorization and authentication

Video timestamp 08:15

Platform as a Service (PaaS)



2 / Getting to Launch

Video timestamp 10:24

Three Stages

— Procure

— Implement

— Authorize

Video timestamp 10:26

Procure

- Pre-procurement sandbox accounts
- IAAs: weeks instead of months
- Pricing: $Risk \times Complexity$
  - Prototyping × Trivial = $20k/ann.
  - FISMA Moderate × Complex = $110k/ann.

Video timestamp 10:52

Pricing

Annual access fee per system

Package	Price
Prototyping	**$15,000**
Open Data	$10,000
FISMA Low	$20,000
FISMA Moderate	$90,000

+

Example resource use for a year

Complexity	Total
Simple	$1000 - $5000
Average	$5000 - $10,000
Complex	$10,000 - $20,000
Epic	$20,000 - $40,000

Video timestamp 11:42

Implement

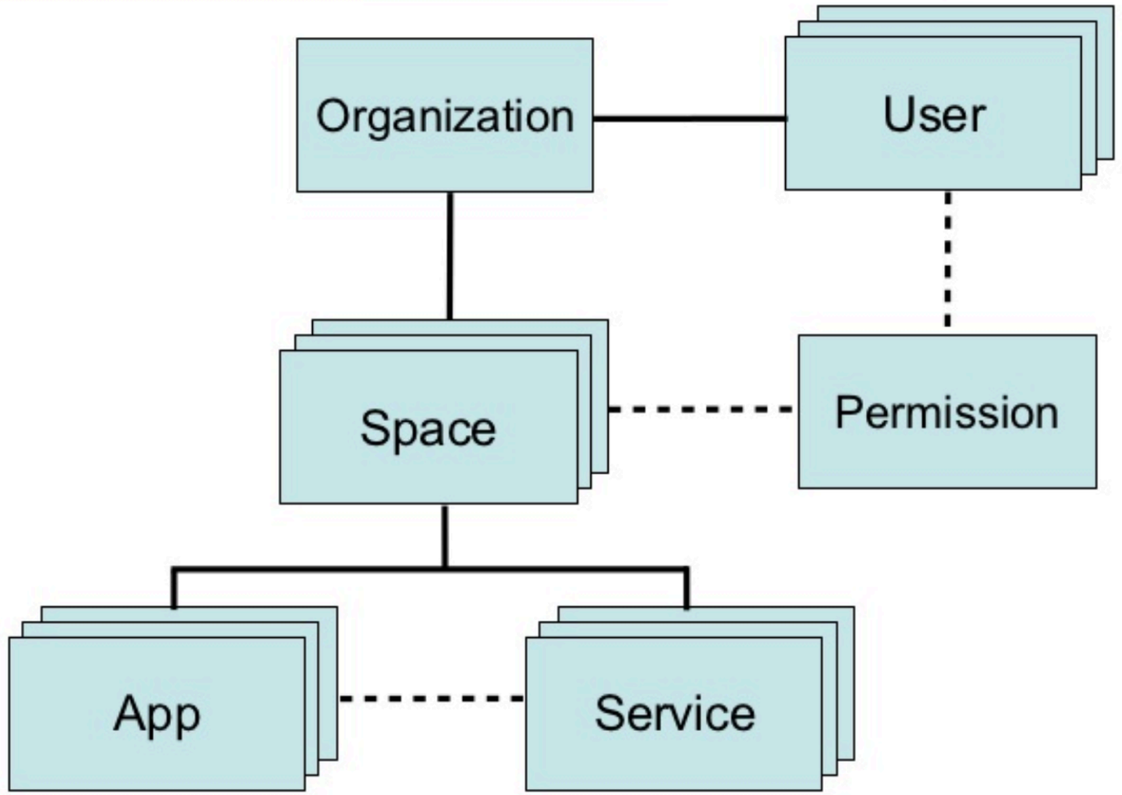
— Users, Spaces & Roles

— Apps

— Services

Video timestamp 13:10

Implement: Users & Roles



- Authentication:
  - Agency IdP or cloud.gov
- Authorization (CF's UAA)
  - Manager, Developer , Auditor ×
  - Organization (EPA, FEC) & Space (dev, stage)

Video timestamp 13:30

Create Role

Step 1 : Set Role Name

Step 2 : Select Role Type

Step 3 : Establish Trust

Step 4 : Attach Policy

Step 5 : Review

Attach Policy

Select one or more policies to attach. Each role can have up to 10 policies attached.

Filter: Policy Type ▾ Readonly **Showing 63 results**

	Policy Name	Attached Entities	Creation Time	Edited Time
☑	ReadOnlyAccess	17	2015-02-07 00:09 UTC+0530	2017-03-21 00:23 UTC+0530
☐	AmazonEC2ReadOnlyAccess	4	2015-02-07 00:10 UTC+0530	2015-02-07 00:10 UTC+0530
☐	AmazonRDSReadOnlyAccess	4	2015-02-07 00:10 UTC+0530	2015-12-17 02:28 UTC+0530
☐	AmazonS3ReadOnlyAccess	3	2015-02-07 00:10 UTC+0530	2015-02-07 00:10 UTC+0530
☐	AmazonSNSReadOnlyAccess	3	2015-02-07 00:11 UTC+0530	2015-02-07 00:11 UTC+0530
☐	AmazonDynamoDBReadOnlyAc...	2	2015-02-07 00:10 UTC+0530	2017-02-27 23:29 UTC+0530
☐	CloudWatchReadOnlyAccess	1	2015-02-07 00:10 UTC+0530	2015-02-07 00:10 UTC+0530
☐	AmazonAppStreamReadOnlyAc...	0	2015-02-07 00:10 UTC+0530	2016-12-08 02:30 UTC+0530
☐	AmazonCloudDirectoryReadOnl...	0	2017-03-01 05:12 UTC+0530	2017-03-01 05:12 UTC+0530
☐	AmazonCognitoReadOnly	0	2015-03-24 22:36 UTC+0530	2016-06-02 23:00 UTC+0530
☐	AmazonEC2ContainerRegistryR...	0	2015-12-21 22:34 UTC+0530	2016-10-12 03:38 UTC+0530
☐	AmazonElastiCacheReadOnlyAc...	0	2015-02-07 00:10 UTC+0530	2015-02-07 00:10 UTC+0530
☐	AmazonElasticFileSystemReadO...	0	2015-05-27 21:55 UTC+0530	2015-05-27 21:55 UTC+0530

Cancel Previous **Next Step**

Video timestamp 15:03

Invite an existing organization user to this space.

Username

--

Add user to this space

Space Managers can change these roles. For details about these roles, see Cloud Foundry roles and permissions.To invite a user and give them roles, see Managing Teammates. ***Removing all roles does not remove a user from an organization. Users with no roles can view other users and their roles while being unable to make any changes.***

peter.burkholder@gsa.gov	☑ Space Developer	☐ Space Manager	☐ Space Auditor	Remove All Space Roles
sys-tester	☑ Space Developer	☑ Space Manager	☐ Space Auditor	Remove All Space Roles
joshua.carp@gsa.gov	☑ Space Developer	☐ Space Manager	☐ Space Auditor	Remove All Space Roles

Video timestamp 15:24

Implement: Dev/Stage/Prod

```
cf create-space dev
cf create-space stage
cf create-space prod
```

Video timestamp 15:47

Implement: Users w/ Spaces

```
cf set-space-role peterb dev SpaceDeveloper
cf set-space-role peterb prod SpaceAuditor
```

Video timestamp 16:02

Time machine

- **Procured ✅**
- **Implemented:**
  - **Users and Authentication ✅**
  - **Dev/Test/Prod Environments ✅**
  - **Roles ✅**

Video timestamp 16:20

Implement: Python Application

```
git clone https://github.com/18F/cf-hello-worlds.git
cd cf-hello-worlds/python-flask
cf push cg-flask-demo
open https://cg-flask-demo.app.cloud.gov
cf scale cg-flask-demo -i 4
```

Video timestamp 17:07

Implement: Language

staticfile	java	ruby
nodejs	go	python
php	binary	dotnet

Video timestamp 17:30

Implement: Services

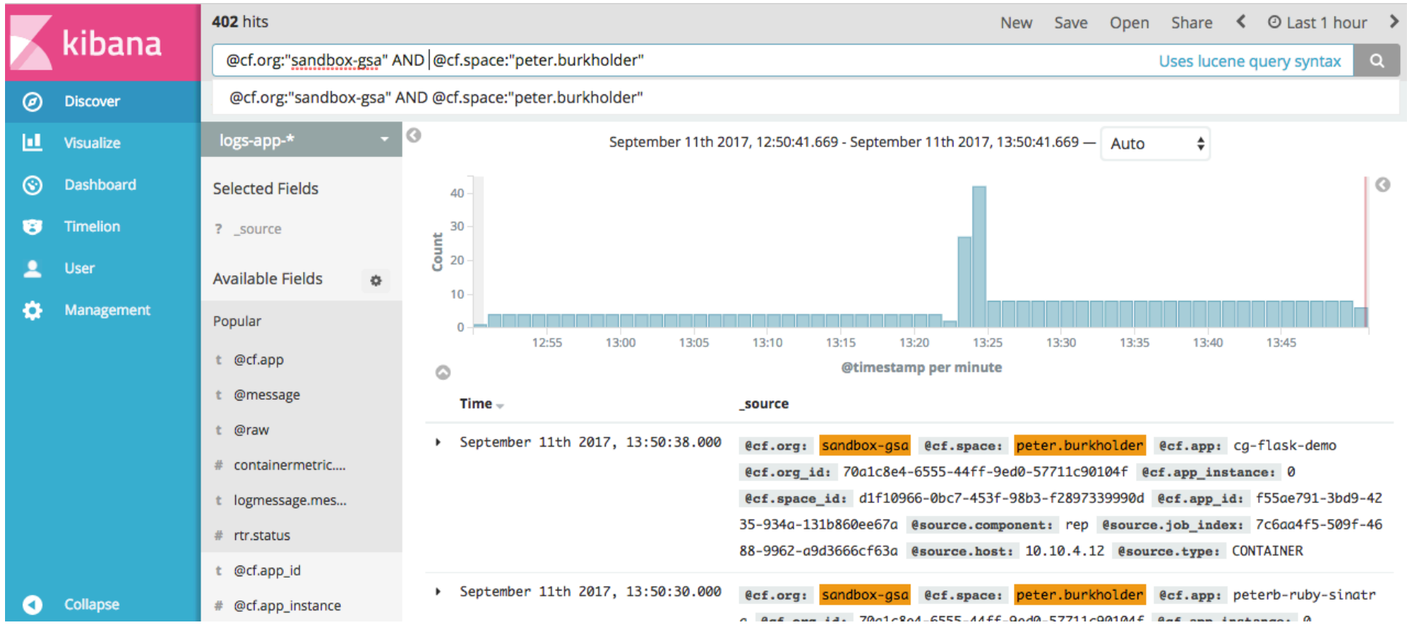
Relational databases (RDS)	PostgreSQL, MySQL, Oracle
Storage (S3)	Private or public data buckets
Custom domain	HTTPS + Content Delivery Network
Redis	In-memory data structure store
Elasticsearch	Full-text search engine
Service accounts	For continuous deployment and auditing
Identity provider	Use cloud.gov authentication in apps

Video timestamp 18:03

Implement: Logs & Diagnostics

— logs: Kibana, custom logdrains

— **cf ssh**: diagnose ephemeral containers



Video timestamp 20:22

Three Stages

— **Procure**

— **Implement**

— Authorize

Video timestamp 20:59

Authorize

- Authority to Operate (ATO)
- Risk Management Framework (Low, Moderate, High)
- NIST 800-53

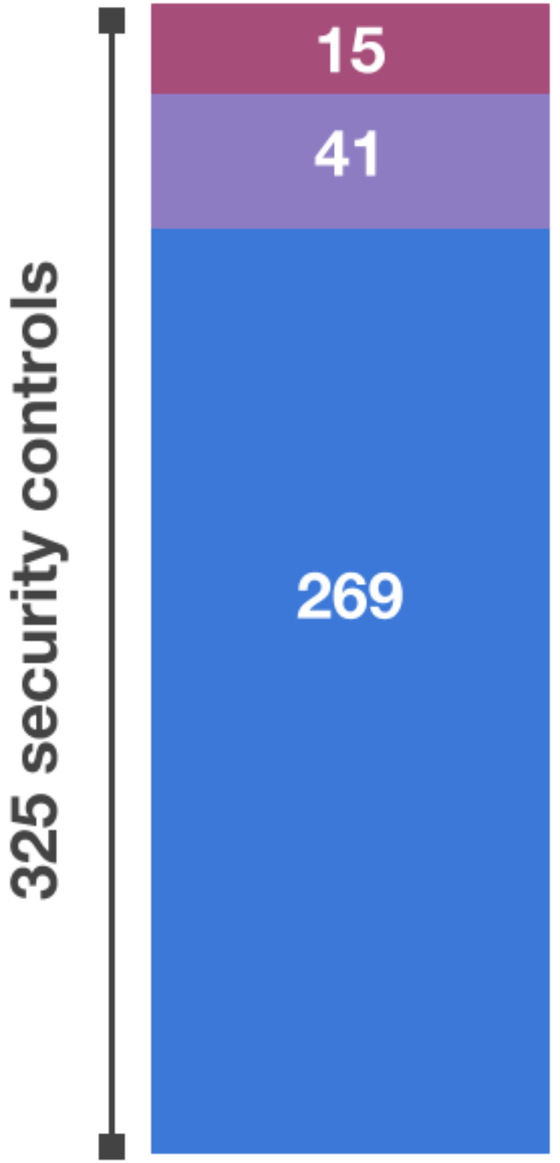
Video timestamp 21:10

Authorize: Controls

- DataCenter: All 325 - You're responsible for:
  - Security Guards, PE-3(3)
  - Disk wiping, MP-6(8)
- IaaS: FedRAMP - You inherit ~88 controls, still 237:
  - System logs, AU-12
  - Kernel patches, SI-2
- cloud.gov:
  - See next slide....

Video timestamp 21:46

Faster ATOs: many controls are handled by cloud.gov



15 customer
IA-2 (12): Accept and verify PIV credentials.

41 shared
IR-9: Information spillage response

269 handled by cloud.gov
SC-39: Separate execution domain for each process

Some will be shared depending on your application.

54 of these are fully inherited from AWS, such as the MA family (maintenance)

Authorize: ATO & Security

— 15 unshared controls, 41 shared
— Simplicity and secure defaults
— Reduce shadow IT (thanks, self-service!)
— Example: **Stack Clash** kernel patch: < 24 hrs

Video timestamp 23:16

Three Stages

— **Procure**

— **Implement**

— **Authorize**

Video timestamp 25:07

Road map features

— TIC ingress control
— PIV/CAC enabled authentication
— App environment security scanning
— Attach a persistent file volume to apps
— AWS resource brokering
— Built-in CI/CD service

Video timestamp 25:15

Let's revisit the mission...

Video timestamp 26:51

~~Suppose~~ Realized

- **A mission**
  - Housing for disaster victims
- **A team**
  - Project / Product Managers
    Designers / Devs
    Ops / Sec
- **A platform**
  - **Build**
  - **Test**
  - **Run**

Video timestamp 26:55

CLOUD.GOV