Fast Forward

Reflecting on a Life of Watching Movies and a Career in Security

Jason Chan
VP, Information Security @ Netflix
@chanjbs
Netflix Subscribers Streamed Record-Breaking 350 Million Hours of Video on Jan. 7

By Janko Roettgers | @janko
So . . . what does this have to do with security?
TRADITIONAL APPLICATION SECURITY

WE HARDLY KNEW YOU...
Back to the movies . . .
Reducing Cognitive Load for Developers
square
/skwər/

noun

1. a plane figure with four equal straight sides and four right angles.
Simplifying the Security Interface for Developers
Are you trying to make your engineers security experts? Or do you just want them to build and operate secure systems?
What security functions can we abstract to simplify the developer experience?
Netflix Studio Engineering

Optimize production from “pitch to play”
Lots of innovation and iteration
Netflix Studio Apps

Netflix Studio User

- Studio LOB App A
- Studio LOB App N
Simplify and Improve Security through Functionality Abstraction
Leverage Netflix OSS - Zuul
“built to enable dynamic routing, monitoring, resiliency and security”

https://github.com/Netflix/zuul/wiki
Netflix Studio Apps with Zuul

Netflix Studio User

Zuul

Pre-Filters
- Rate Limit
- IP Blacklist
- Authentication
- Authorization
- SigSci WAF
- Schema Check

Post-Filters
- Schema Check
- Sec Headers
- DLP

Studio LOB App A
- Studio LOB App N
Results

Lower cognitive load for onboarding security
Centralized and managed functionality

_Frees developers to build the Netflix Studio!_
Blurring Lines: App and Infra

Monolith to microservices: network
Immutable infra: OS, custom app, middleware
Infra as code: Everything!
Tackling App and Infra Integration: Seamless Least Privilege
The Magic of IaaS

- PostgreSQL
  - Database
- Hadoop
  - Storage
- Email Services
- Message Queue
  - Instances
- Hadoop
AWS access control is via the IAM (Identity and Access Management) service.

Applications use roles.

Roles have attached policies.

Policies have permissions.
Ex: Cloud Based Word Processor
Ex: Cloud Based Word Processor

{  
"Effect": "Allow",  
"Action": ["*:*"],  
"Resource": "*"
}

Ex: Cloud Based Word Processor

{  
  "Effect": "Allow",
  "Action": ["s3:*"],
  "Resource": "*"
}

Ex: Cloud Based Word Processor

{
  "Effect": "Allow",
  "Action": ["s3:GetObject", "s3:PutObject"],
  "Resource": "*
}

Ex: Cloud Based Word Processor

{
  "Effect": "Allow",
  "Action": ["s3:GetObject", "s3:PutObject"],
  "Resource": "arn:aws:s3:::wp_bucket"
}
AWS provides data about API use

This data acts as a basis for action
When a new application is created, we provide a base set of permissions:

- s3:GetObject
- s3:PutObject
- ...
- ...
- ...
- ...
- sqs:ReceiveMessage
We observe the application to see which permissions are actually used.
We then remove unused permissions

s3:GetObject
s3:PutObject
...
...
...
...
sqs:ReceiveMessage
We then remove unused permissions

- s3:GetObject
- s3:PutObject
- sqs:SendMessage
- sqs:ReceiveMessage
Available as OSS - Repokid

https://github.com/Netflix/repokid
Results

Low-risk access reduction
Transparent and versioned ops
Innovation and high-velocity development without friction
Potential for “Controlled” Anarchy

Microservices
YBIYRI
Polyglot and multiple tech stacks
Independent deployments

*Intentionally decentralized governance leads to increased attack surface*
Managing the Anarchy
Prefer Self-Service and Make Security Expectations Explicit and Measurable
**Prod Ready**
- SRE-driven developer outreach
- Evangelize successful patterns and practices
- Automated scoring
- Uncover risk and reward operational excellence

**Paved Road**
- Well-supported solutions from central teams
- Addresses many common technical areas
- “Technology Standards”
Security Prod Ready (ex.)

Per-app IAM role
Per-app Security Group
No secrets in code
Instance identity
Updated machine image
Production Ready Scorecard

Prod Ready Score

77%

- alerts: 85%
- autoscaling: 91%
- canary: 0%
- chaos: 0%
- dashboards: 86%
- ezconfig: 86%

Criterion: autoscaling
Week of Date: August 28, 2016
Score: 91%

Tracking over time
Paved Road (ex.)

Quarterly Change Cycle

Commit to update once per quarter to pull in upgrades, library changes, and modifications to paved road components
Deprecations and Blacklists

**gandalf-agent <=0.16.0**

**STARTED ON APR 10, 2018**
**BY PLATFORM SECURITY**
**PLATFORMSECURITY@NETFLIX.COM**

---

**Server Group Usage History**

- Apr 13 at 6:00 pm: 67 server groups
**Intention** - Most security backlog is standard/commodity; explicitly limit bespoke/custom backlog
Results

Security expectations are clear and measurable
Paved road provides a path of convergence
Self-service for customers and providing teams provides a virtuous cycle
In closing . . .
Overall Takeaways

Stay attuned to trends
Simplify and standardize
Favor transparent decisions
Measure adoption and uptake
Get comfortable with tradeoffs
Thank you SANS Summit!

DIRECTOR
Jason Chan

DATE
10/22

SCENE
Finale

TAKE
1

CAMERA
@chanibs