

# Insecure Transit

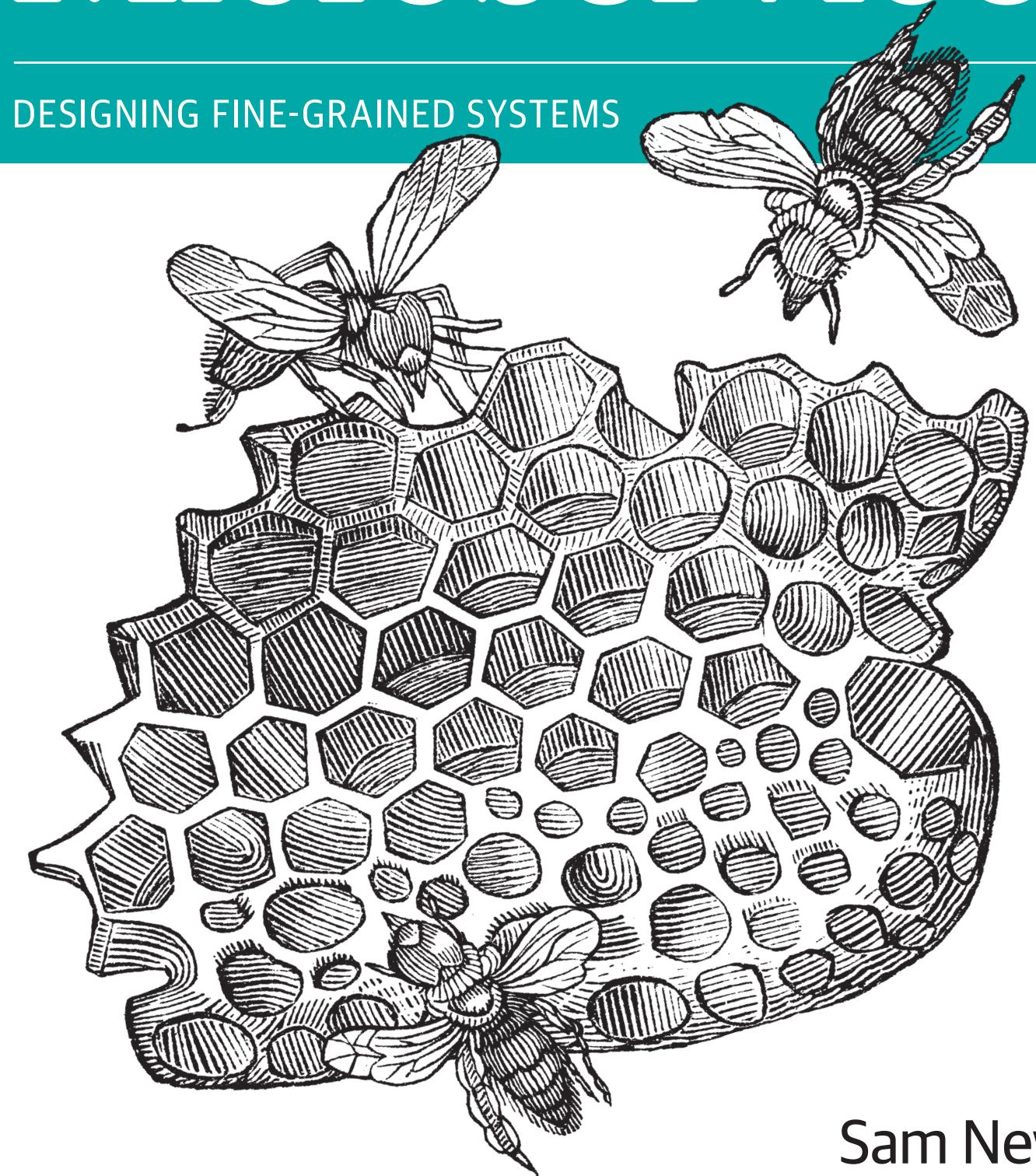
Microservice Security

Sam Newman - QCON London 2018

O'REILLY®

# Building Microservices

DESIGNING FINE-GRAINED SYSTEMS



Sam Newman

Sam Newman &  
Associates

# Massive Equifax data breach - what you need to know



By **Callum Mason**, News Reporter  
12 Sep 2017 | Updated 18 Sep 2017



Credit report heavyweight Equifax has warned that up to 400,000 UK consumers may have had their personal details stolen as part of a massive global data breach. Info on exactly who's been affected and what you can do about it is still somewhat sketchy, but here's what we know.

Equifax revealed on 8 September that 143 million consumers in the US could have been affected by the incident, which saw hackers access data such as names, address and dates of birth, as well as credit card numbers in a smaller number of cases.

Although its UK business – Equifax Ltd – now says systems in this country are not affected, it admits a file which was stored in the US and contained more limited personal information on up to 400,000 UK consumers may have been accessed.

## Related MSE Guides

[Credit Scores](#)  
Bust myths & improve your score

[30+ Ways to Stop Scams](#)  
As scams get clever, we need to too!

[Check your credit report for free](#)  
Grab your file and check your score, or even get PAID to do it



### Get Our Free Money Tips Email!

For all the latest deals, guides and loopholes - join the 12m who get it.  
*Don't miss out*

Enter Email Address

GET IT!

[FAQs](#) | [Privacy Policy](#) | [Past Emails](#) | [Unsubscribe](#)

## What is Equifax and what data does it have?

Equifax is the second biggest credit reference agency in the UK, after Experian.

<https://www.moneysavingexpert.com/news/protect/2017/09/massive-equifax-data-breach---what-you-need-to-know>

@samnewman

Security

## Meltdown, Spectre: The password theft bugs at the heart of Intel CPUs

AMD, Arm also affected by data-leak design blunders, Chipzilla hit hardest

By Chris Williams, Editor in Chief 4 Jan 2018 at 07:29

252 SHARE ▾



**Summary** The severe design flaw in Intel microprocessors that allows sensitive data, such as passwords and crypto-keys, to be stolen from memory is real – and its details have been revealed.

On Tuesday, we warned that a [blueprint blunder in Intel's CPUs](#) could allow applications, malware, and JavaScript running in web browsers, to obtain information they should not be allowed to access: the contents of the operating system kernel's private memory areas. These zones often contain files cached from disk, a view onto the machine's entire physical memory, and other secrets. This should be invisible to normal programs.

[https://www.theregister.co.uk/2018/01/04/intel\\_amd\\_arm\\_cpu\\_vulnerability/](https://www.theregister.co.uk/2018/01/04/intel_amd_arm_cpu_vulnerability/)

@samnewman



## GDPR Portal: Site Overview

This website is a resource to educate the public about the main elements of the General Data Protection Regulation (GDPR).

After four years of preparation and debate the GDPR was finally approved by the EU Parliament on 14 April 2016. Enforcement date: 25 May 2018 - at which time those organizations in non-compliance may face heavy fines.

The EU General Data Protection Regulation (GDPR) replaces the Data Protection Directive 95/46/EC and was designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens data privacy and to reshape the way organizations across the region approach data privacy. The key articles of the GDPR, as well as information on its business impact, can be found throughout this site.

## Quick Links

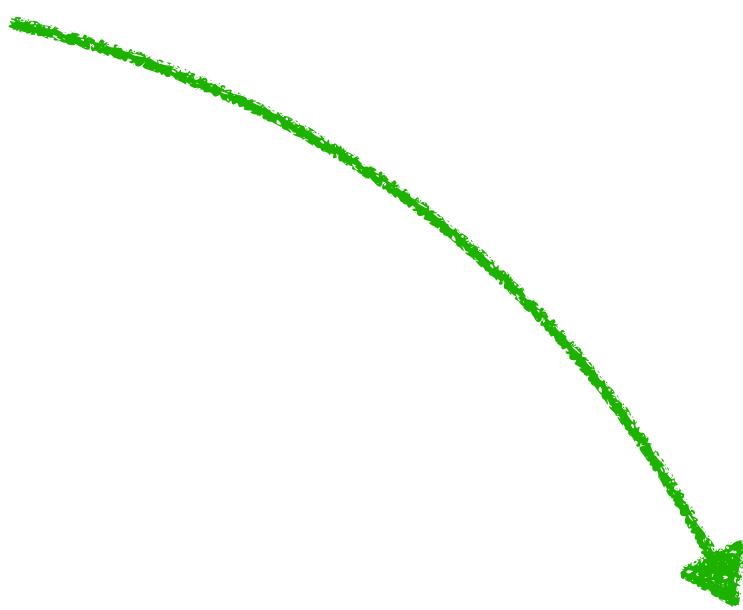
[GDPR Key Changes](#)  
Summary of key changes

[FAQs](#)  
How to prepare?  
Is my organization affected?  
What does Brexit mean for GDPR?

<https://www.eugdpr.org>

# **Design**

**Design**

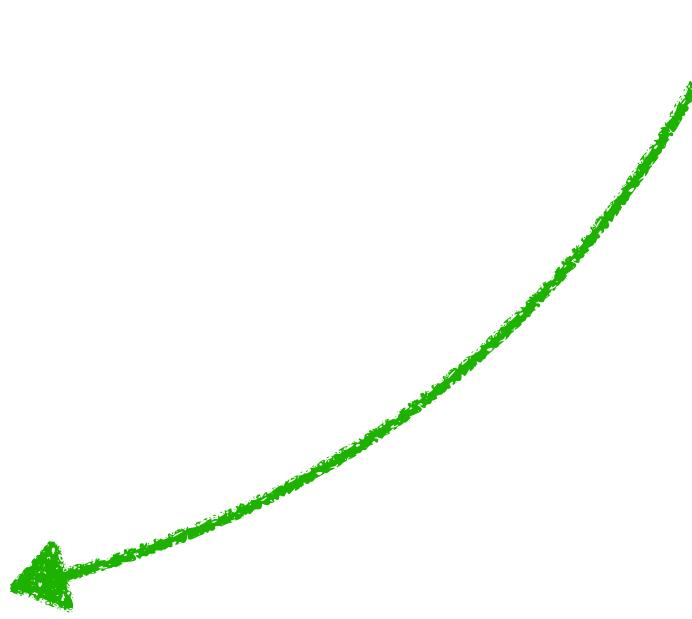
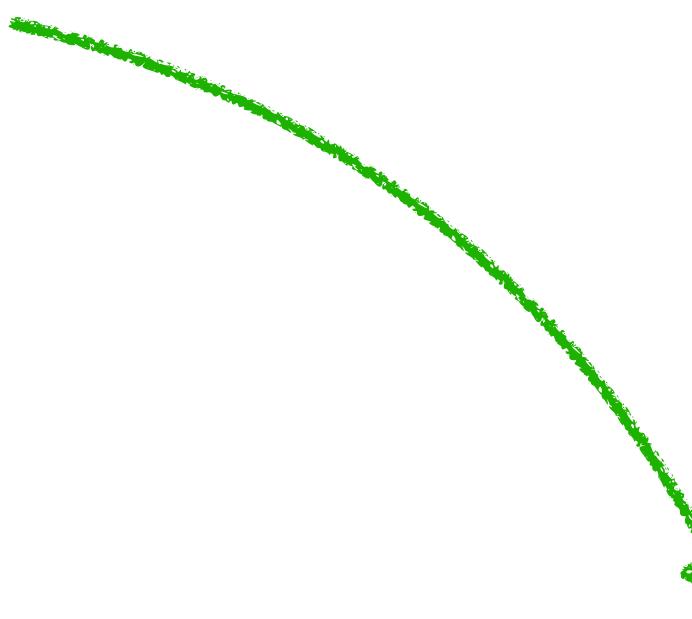


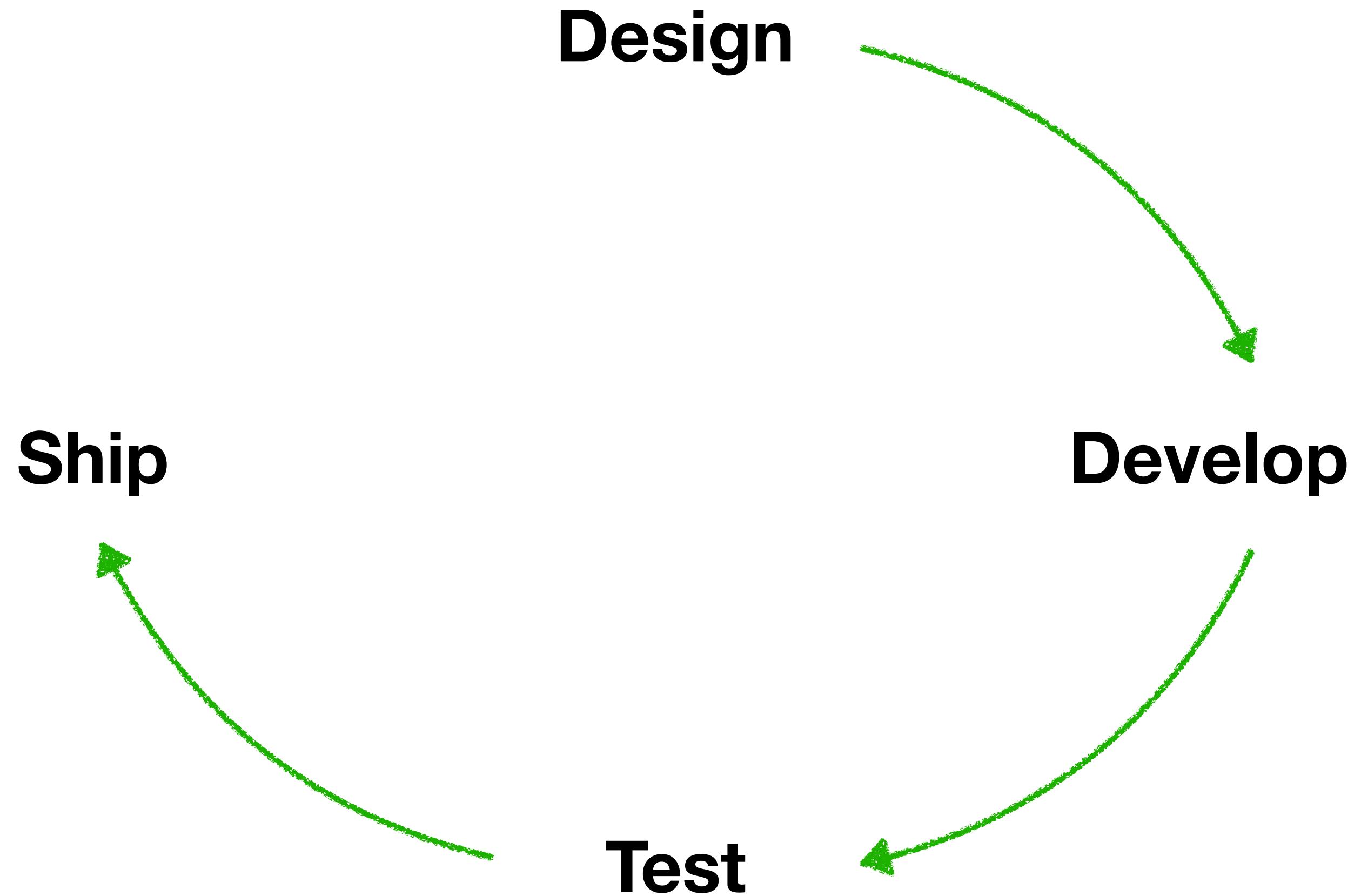
**Develop**

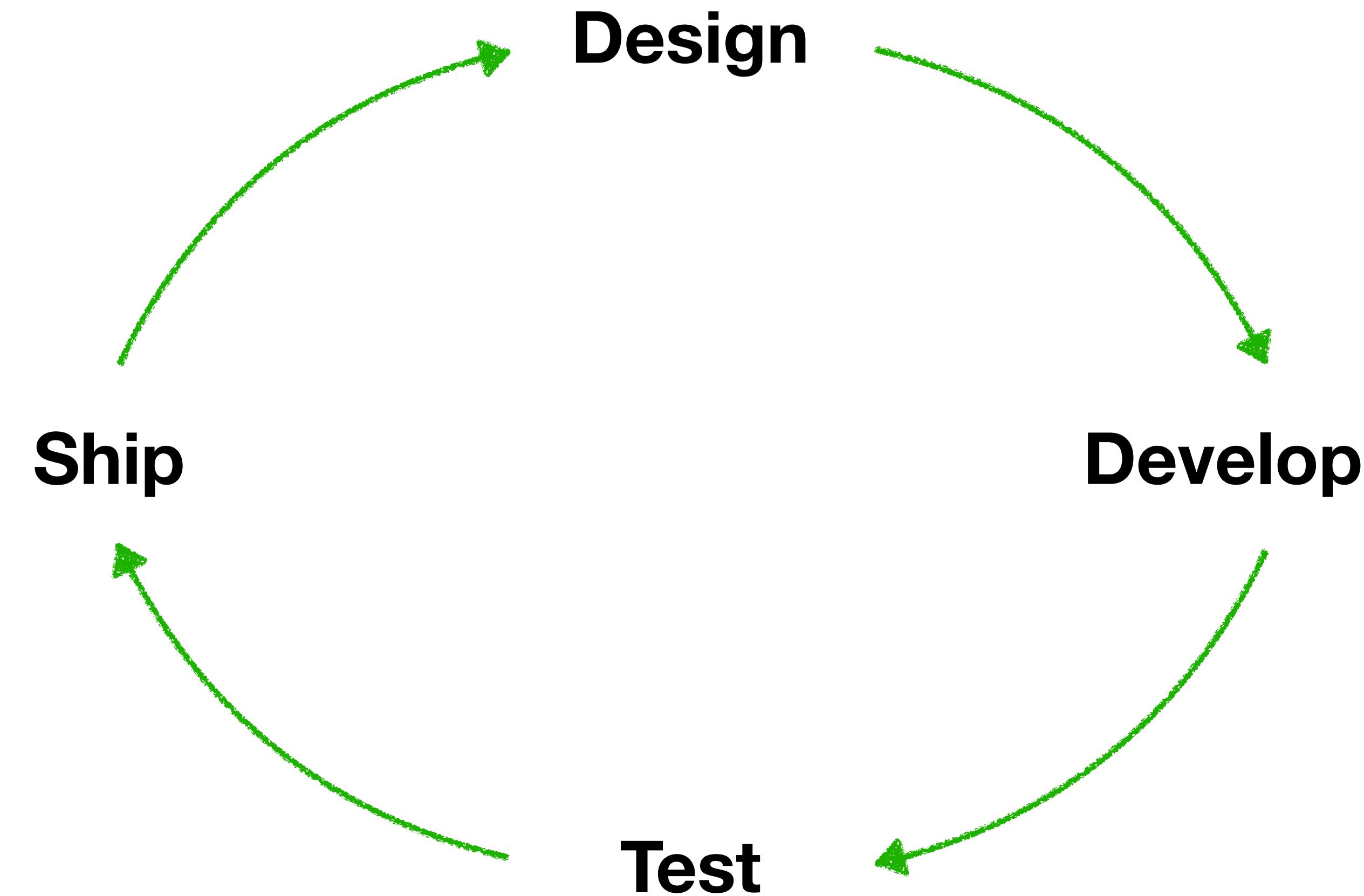
**Design**

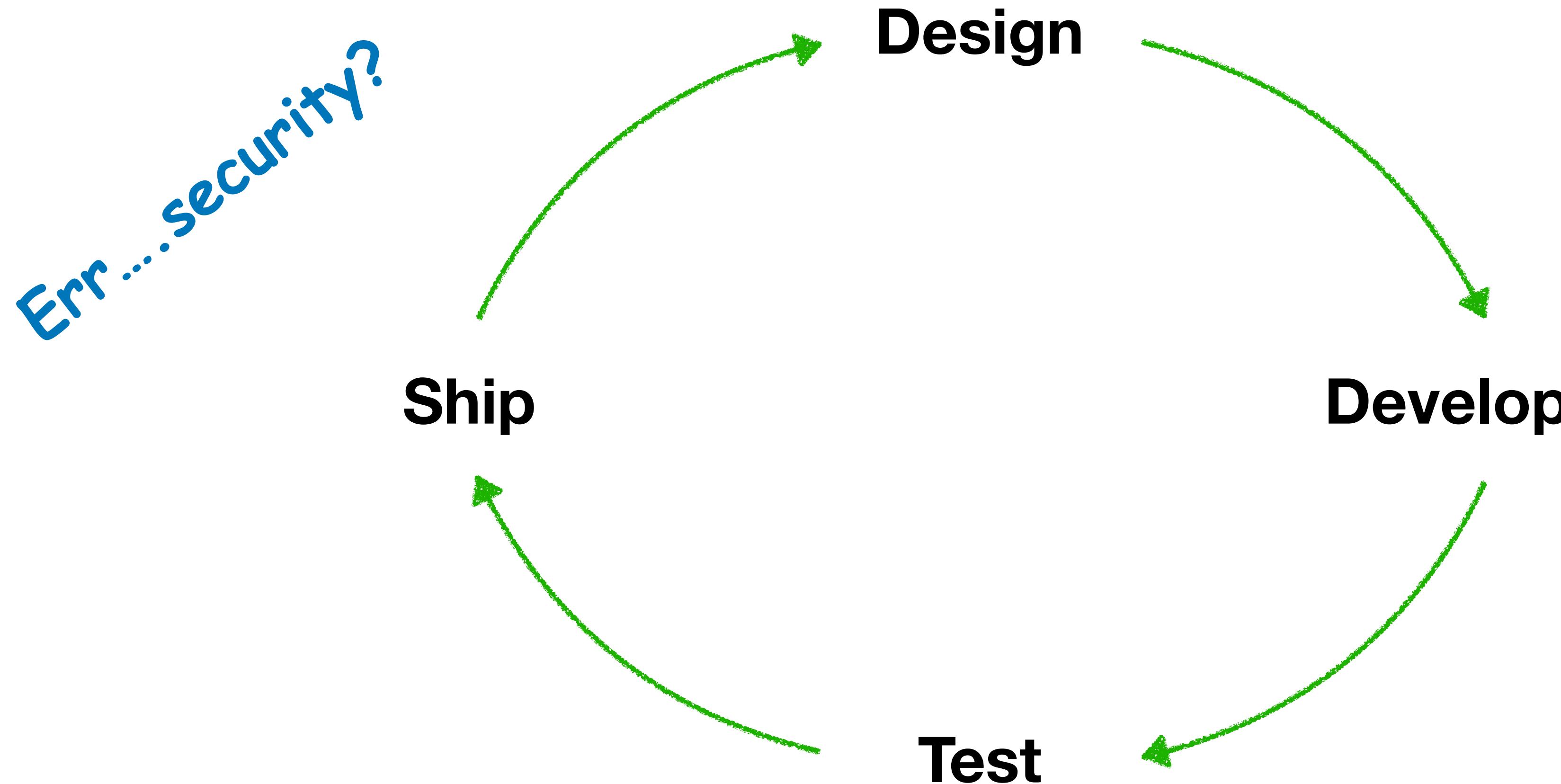
**Develop**

**Test**











<https://www.flickr.com/photos/labyrinthx/195473339/>





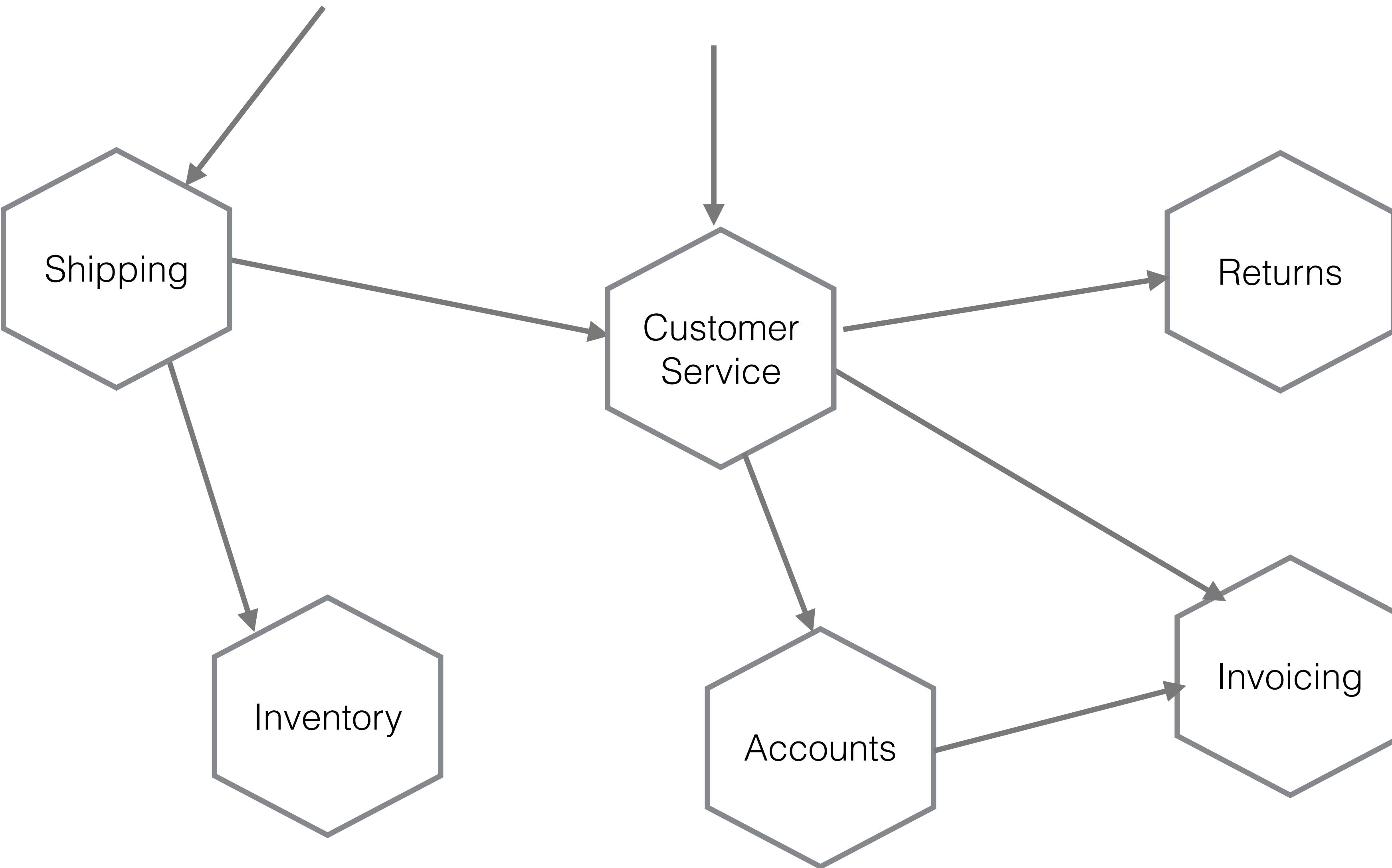
<https://www.flickr.com/photos/seattlemunicipalarchives/4058808950>

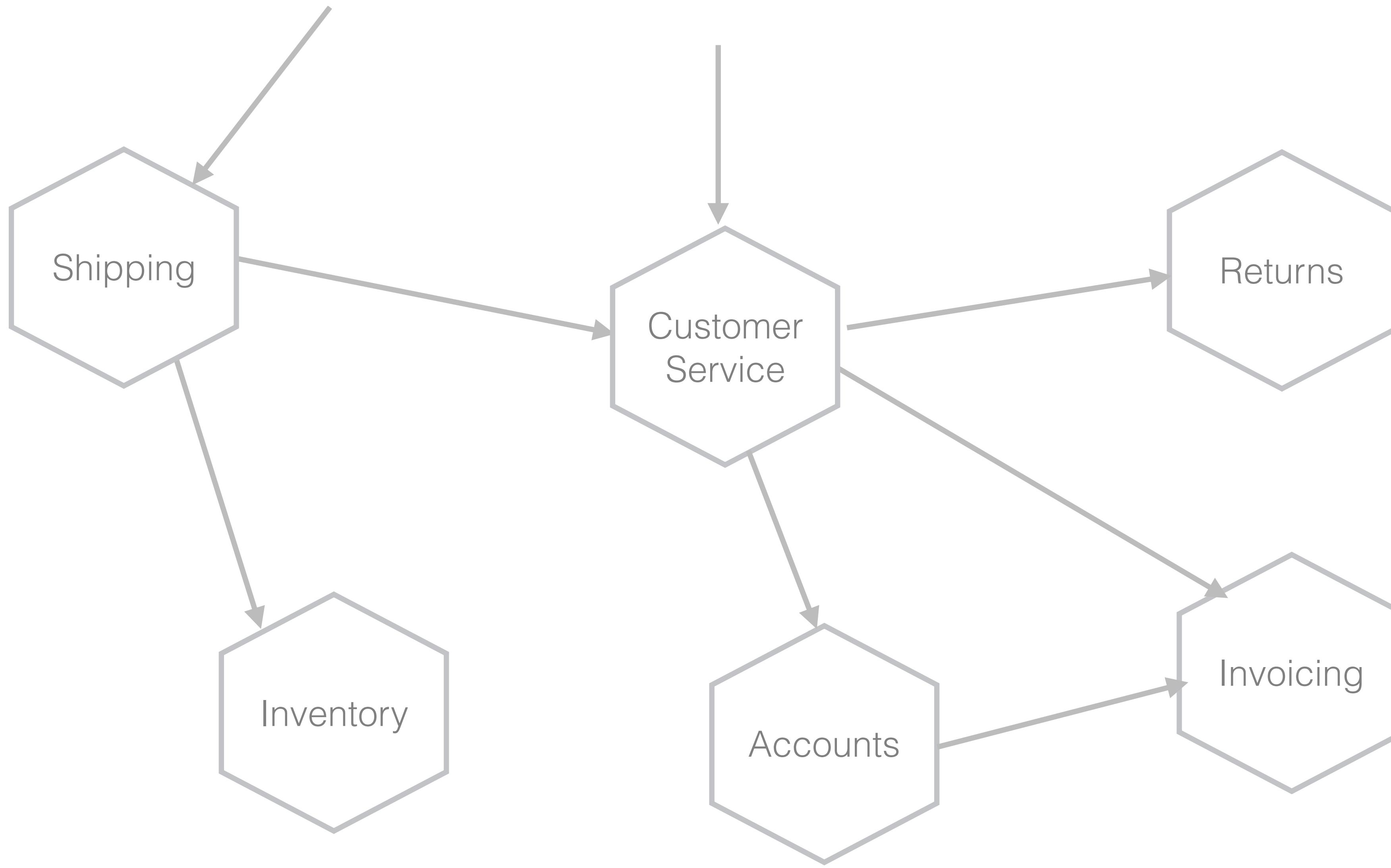
amnewman

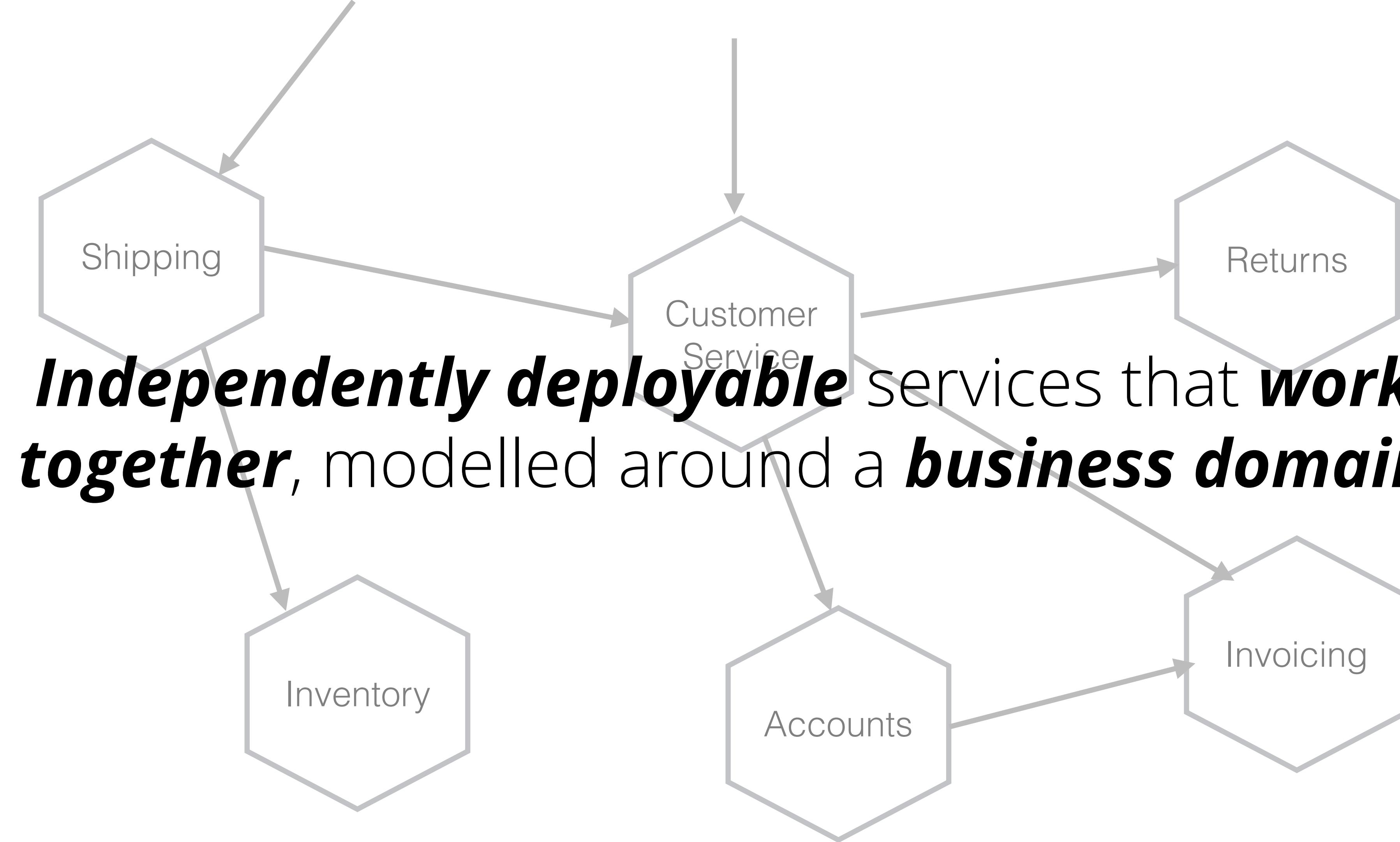


<https://www.flickr.com/photos/theseanster93/485390997/>

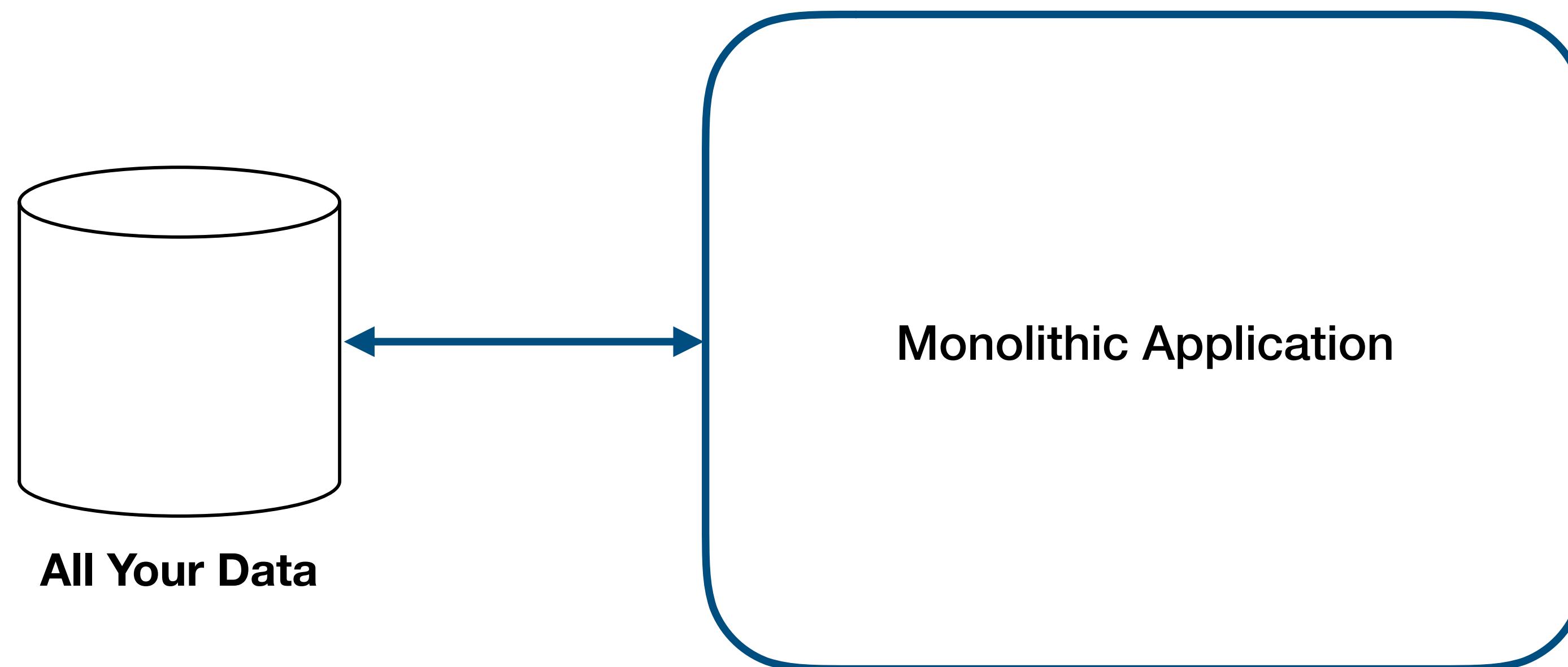
# Just Enough Security







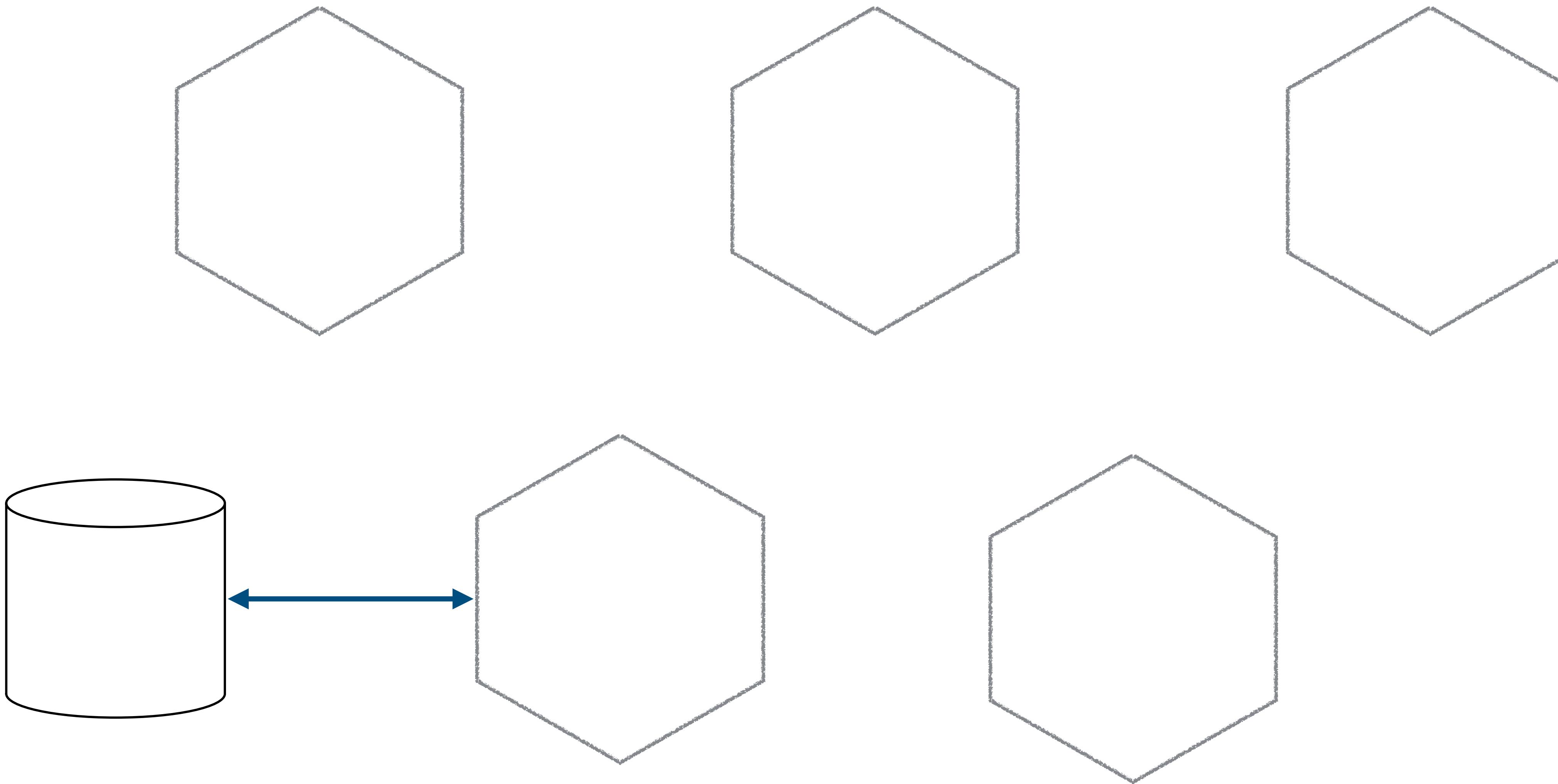


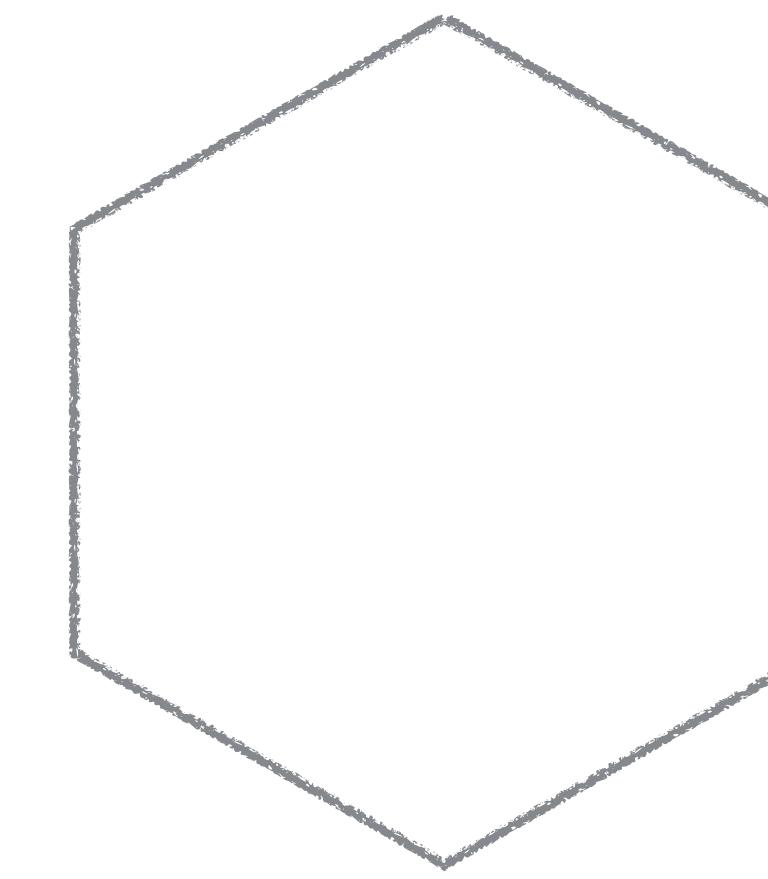
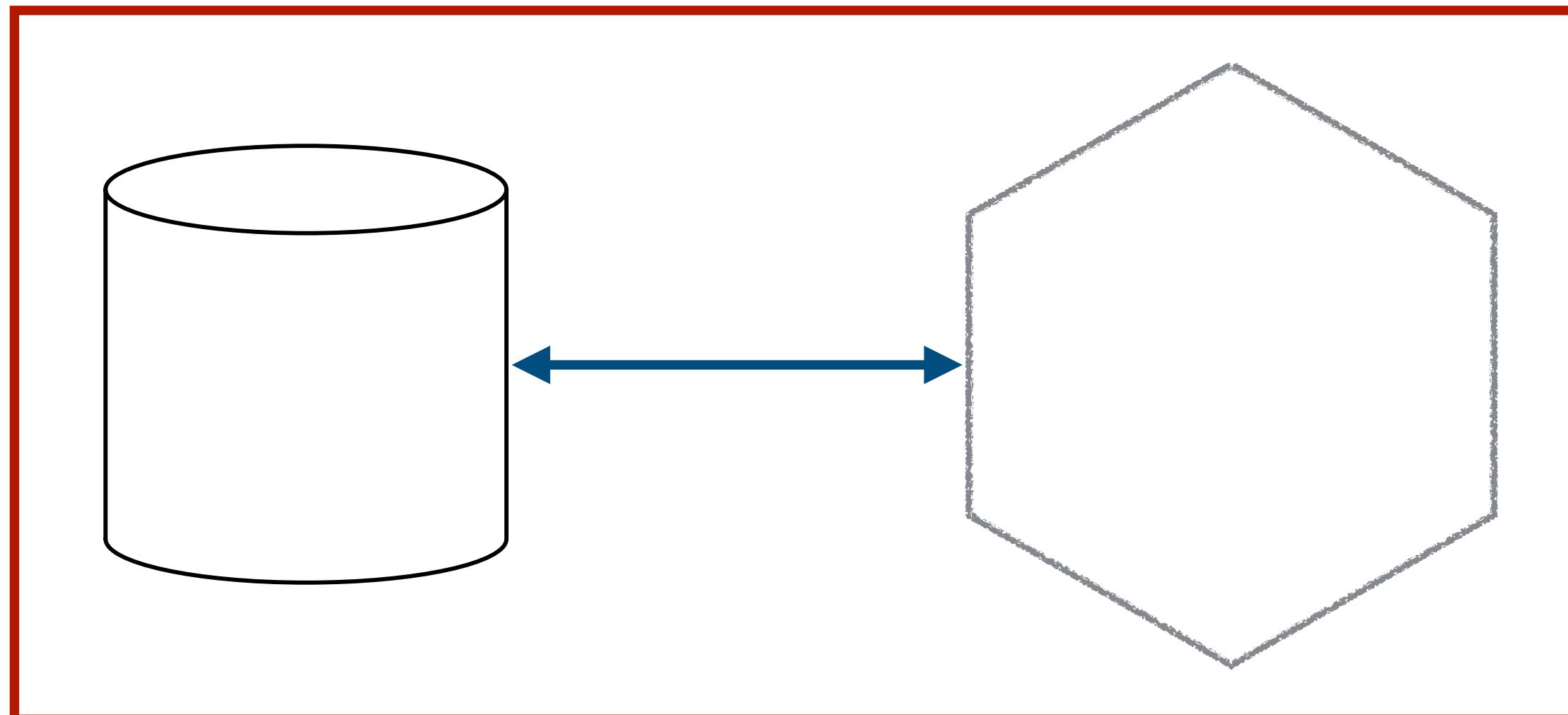
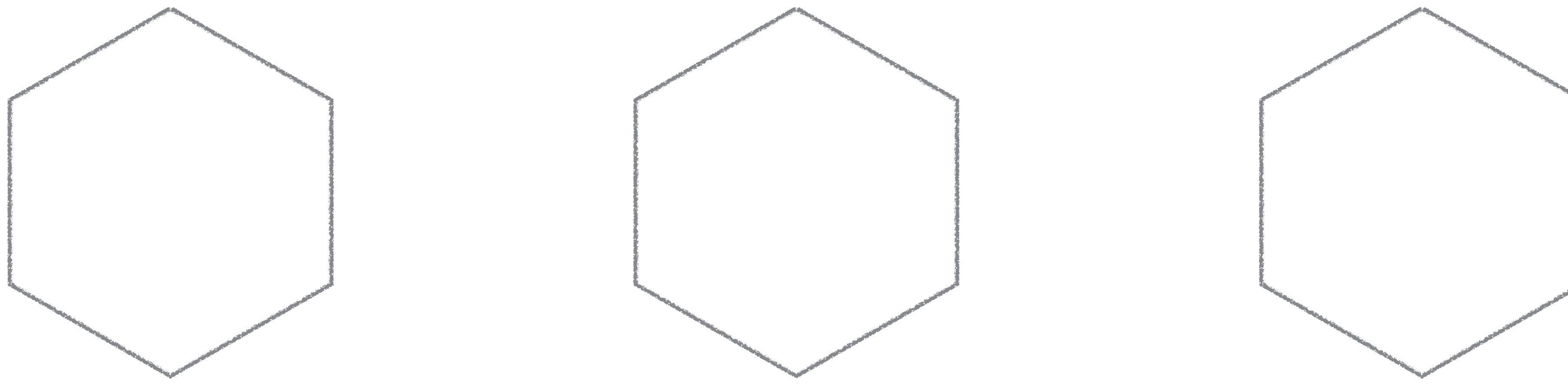




<https://www.flickr.com/photos/lkowen/15803718243/>

@samnewman





# Guide to the General Data Protection Regulation (GDPR)

Share



Download options



Search this document



Introduction

What's new

Key definitions

Principles

Lawful basis for processing

Consent

Legitimate interests

Special category data

Criminal offence data

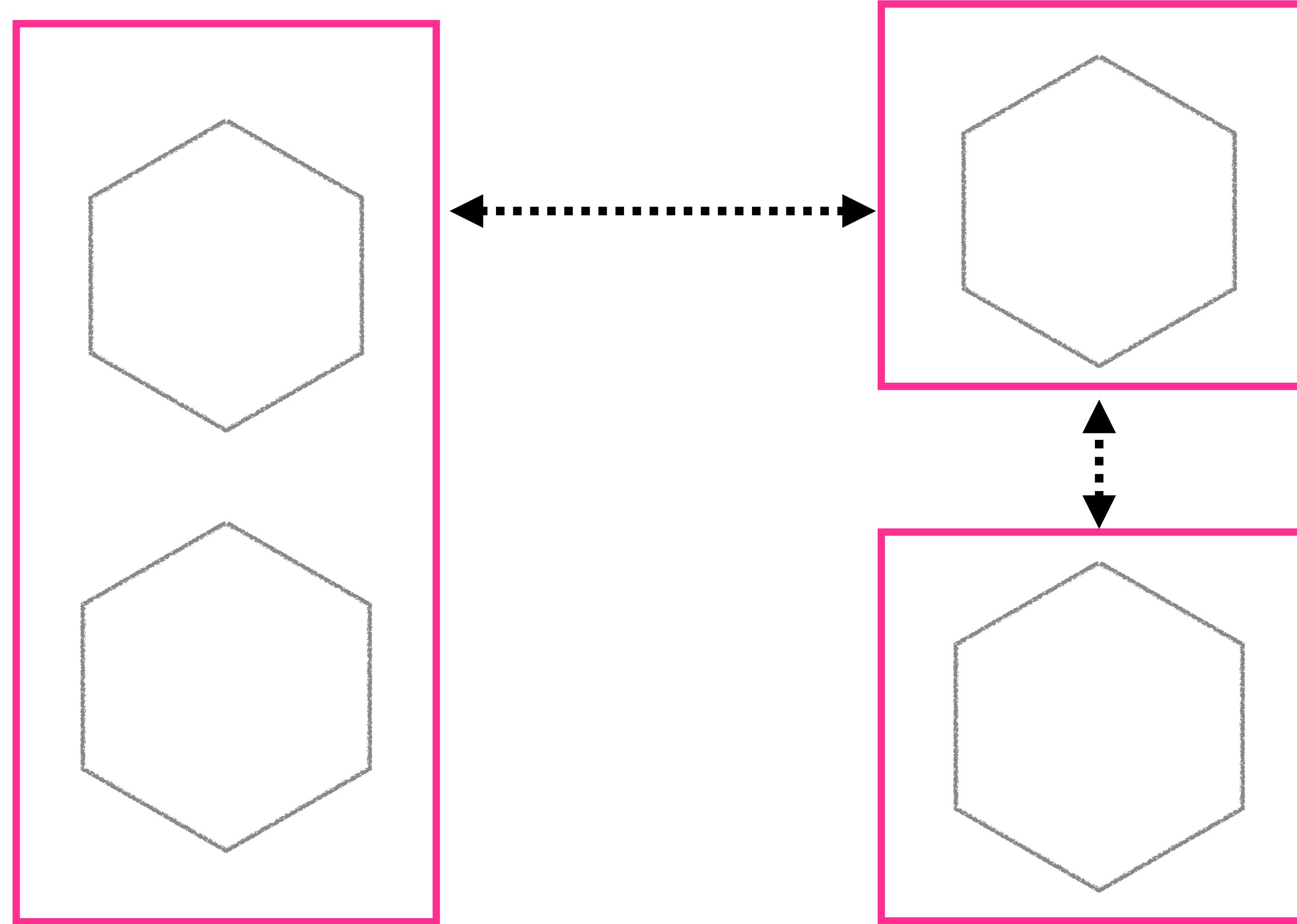
## Introduction

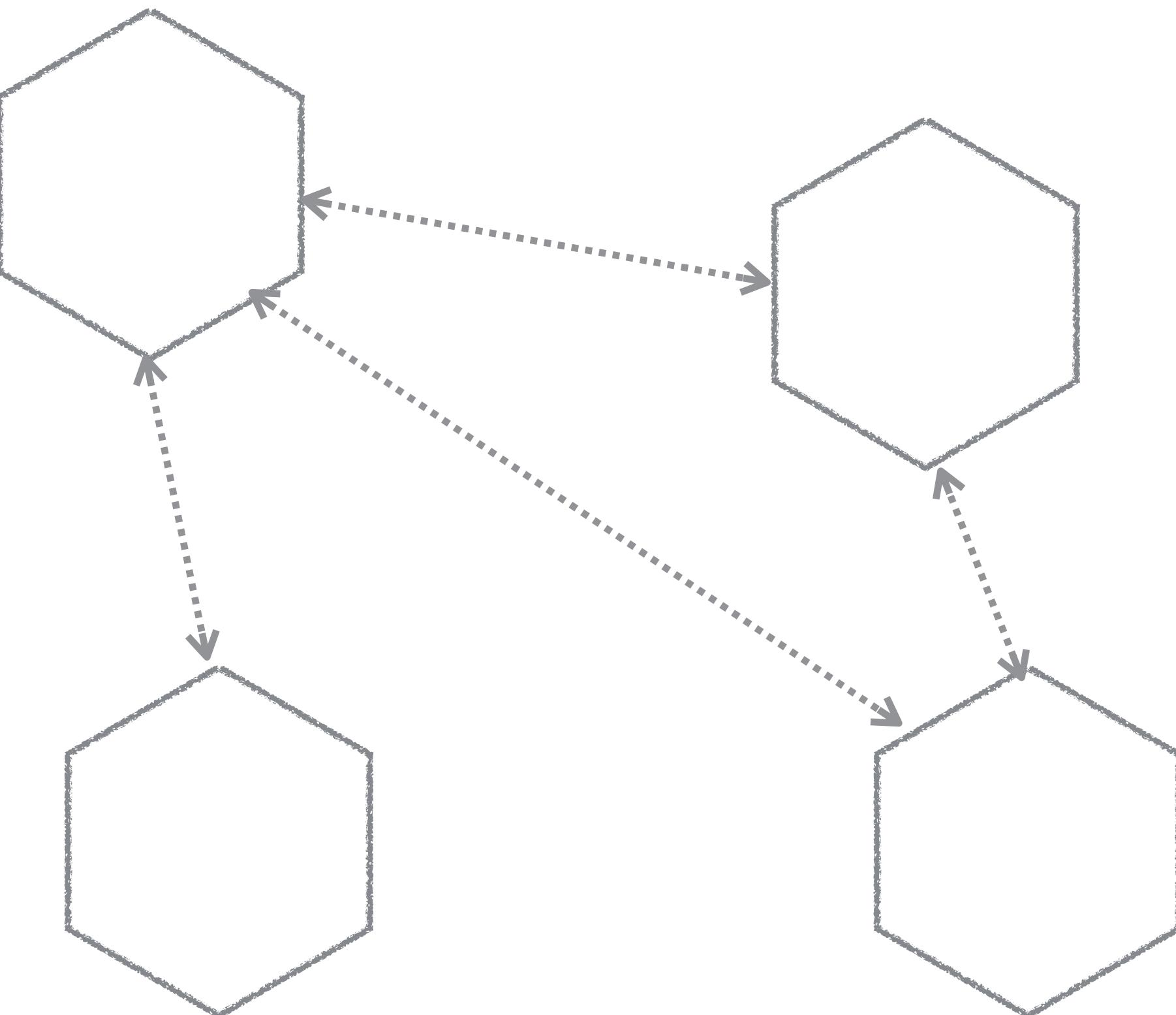
The Guide to the GDPR explains the provisions of the GDPR to help organisations comply with its requirements. It is for those who have day-to-day responsibility for data protection.

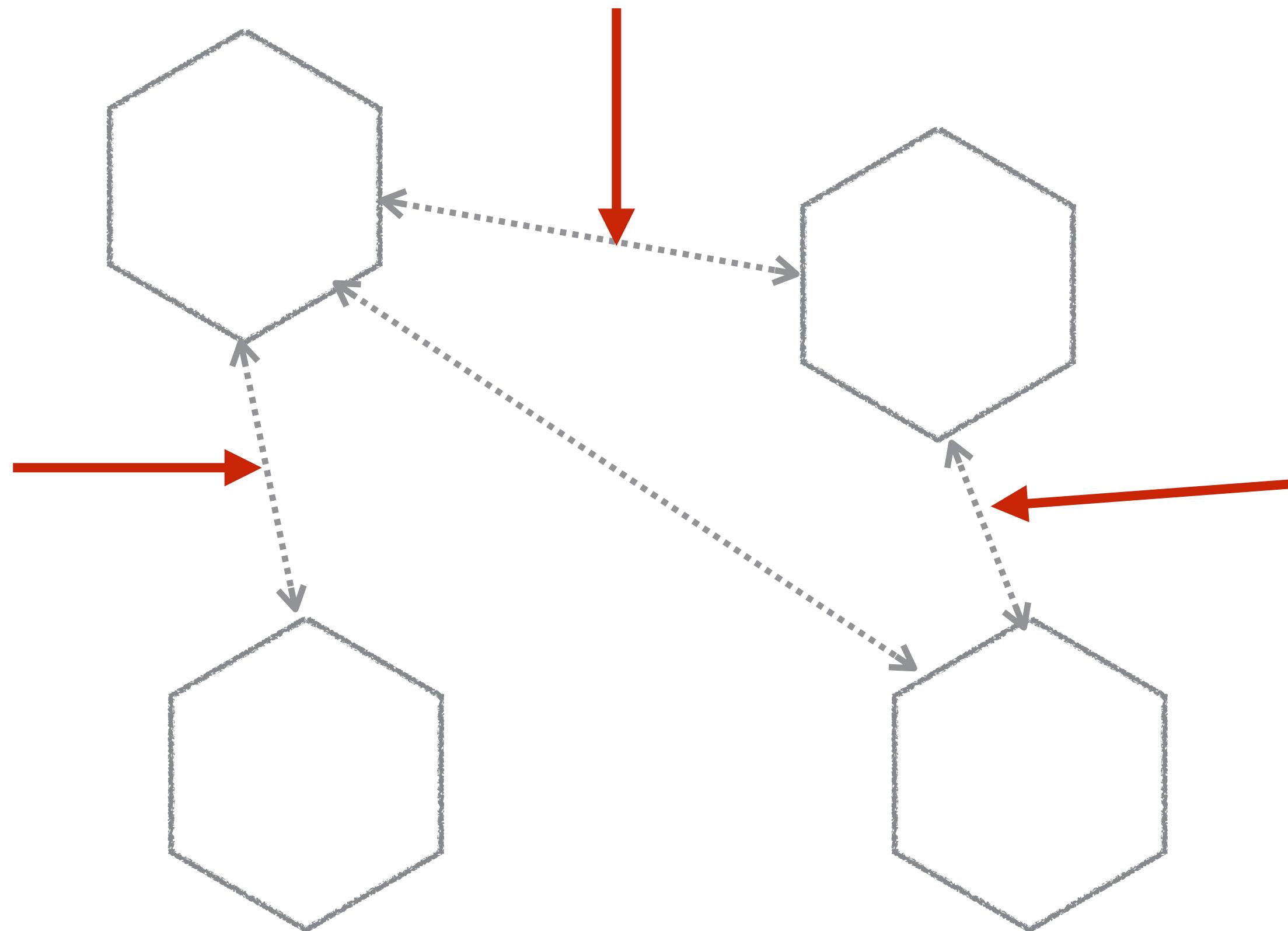
This is a living document and we are working to expand it in key areas. It includes links to relevant sections of the GDPR itself, to other ICO guidance and to guidance produced by the EU's Article 29 Working Party. The Working Party includes representatives of the data protection authorities from each EU member state, and the ICO is the UK's representative.

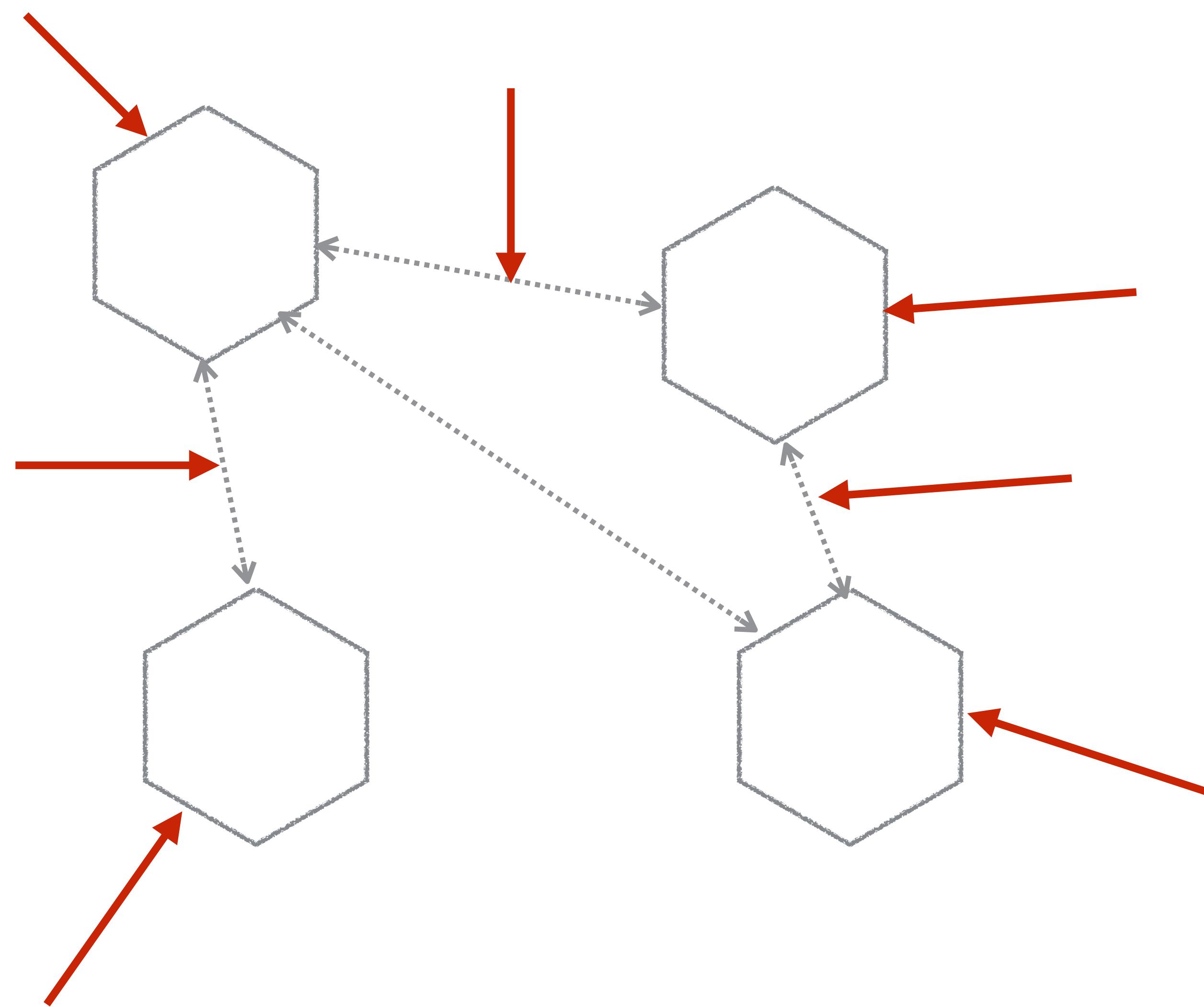
Alongside the Guide to the GDPR, we have produced a number of tools to help organisations to prepare for the GDPR:

[GDPR: 12 steps to take now](#)









# The Basics

**Who here thinks they can assess risks?**



[http://www.verizonenterprise.com/resources/reports/rp\\_DBIR\\_2017\\_Report\\_en\\_xg.pdf](http://www.verizonenterprise.com/resources/reports/rp_DBIR_2017_Report_en_xg.pdf)

@samnewman

## HOW DO BREACHES OCCUR?



<http://www.verizonenterprise.com/verizon-insights-lab/dbir/2017/>

## HOW DO BREACHES OCCUR?



<http://www.verizonenterprise.com/verizon-insights-lab/dbir/2017/>

# BETTER PASSWORD RULES?

## Passwords Evolved: Authentication Guidance for the Modern Era



25 JULY 2017

In the beginning, things were simple: you had two strings (a username and a password) and if someone knew both of them, they could log in. Easy.

But the ecosystem in which they were used was simple too, for example in [MIT's Time-Sharing Computer](#), considered to be the first computer system to use passwords:



<https://www.troyhunt.com/passwords-evolved-authentication-guidance-for-the-modern-era/>

## BETTER PASSWORD RULES?

### Passwords Evolved: Authentication Guidance for the Modern Era



25 JULY 2017

In the beginning, things were simple: you had two strings (a username and a password) and if someone knew both of them, they could log in. Easy.

But the ecosystem in which they were used was simple too, for example in [MIT's Time-Sharing Computer](#), considered to be the first computer system to use passwords:



**Summarises ideas from NIST and the UK's National Cyber Security Centre**

<https://www.troyhunt.com/passwords-evolved-authentication-guidance-for-the-modern-era/>

## BETTER PASSWORD RULES?

Passwords Evolved: Authentication Guidance for the Modern Era

25 JULY 2017

In the beginning, things were simple: you had two strings (a username and a password) and if someone knew both of them, they could log in. Easy.

But the ecosystem in which they were used was simple too, for example in [MIT's Time-Sharing Computer](#), considered to be the first computer system to use passwords:



Summarises ideas from NIST and the UK's National Cyber Security Centre

Packed with great tips, like...

<https://www.troyhunt.com/passwords-evolved-authentication-guidance-for-the-modern-era/>

# PASSWORDS EVOLVED

<https://www.troyhunt.com/passwords-evolved-authentication-guidance-for-the-modern-era/>

@samnewman

# PASSWORDS EVOLVED

Longer is stronger

# PASSWORDS EVOLVED

Longer is stronger

Eliminate complex character composition rules



<https://www.pinterest.dk/pin/566679565591724157/>

# PASSWORDS EVOLVED

Longer is stronger

Eliminate complex character composition rules

# PASSWORDS EVOLVED

Longer is stronger

Eliminate complex character composition rules

Embrace password managers

# PASSWORDS EVOLVED

Longer is stronger

Eliminate complex character composition rules

Embrace password managers

Do not mandate password changes

## PASSWORDS EVOLVED

Longer is stronger

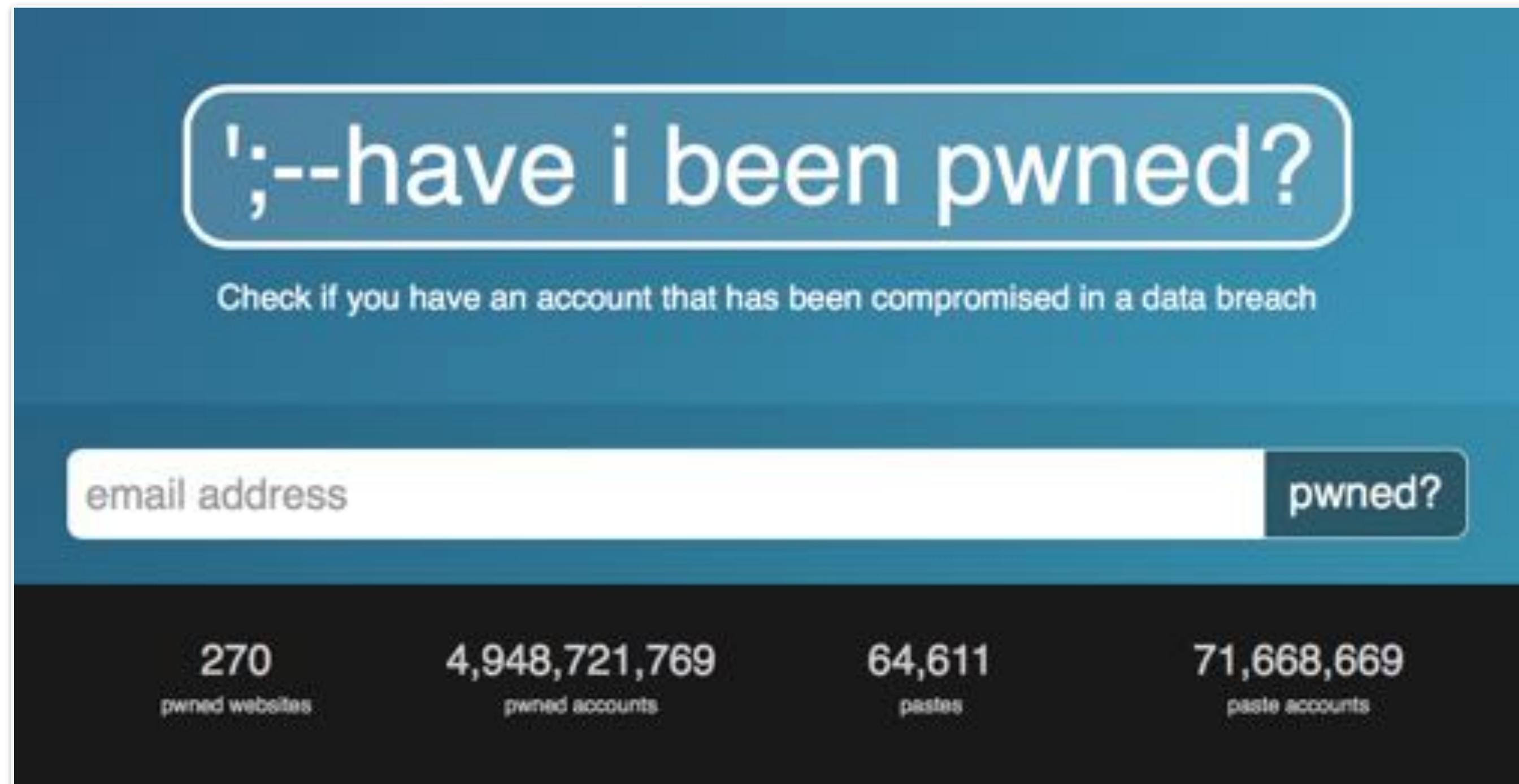
Eliminate complex character composition rules

Embrace password managers

Do not mandate password changes

Block previously breached passwords

## HAVE I BEEN PWNED?



<https://haveibeenpwned.com>

@samnewman

# HAVE I BEEN PWNED?

The screenshot shows the Have I Been Pwned? website interface. At the top, there's a large blue header with the text '';--have i been pwned?' in white. Below the header, a sub-header says 'Check if you have an account that has been compromised in a data breach'. On the left, there's a form field labeled 'email address' containing 'sam.newman@gmail.com'. To the right of the email field is a button labeled 'pwned?'. On the far left, there's a sidebar with two statistics: '270' under 'pwned websites' and '4,948,721,769' under 'pwned accounts'. The main content area is red and displays the message 'Oh no — pwned!' in white. Below this, it says 'Pwned on 13 breached sites and found 1 paste (subscribe to search sensitive breaches)'. At the bottom of the red area, there are links for 'Notify me when I get pwned' and 'Donate', along with social media icons for Facebook, Twitter, and LinkedIn.

<https://haveibeenpwned.com>

@samnewman

# CHECK FOR BREACHED PASSWORDS!

Downloading the Pwned Passwords list					
	File	Date	Size	Description	SHA-1 hash of 7-Zip file
<a href="#">torrent</a> <a href="#">Cloudflare</a>	Version 2 (ordered by prevalence)	22 Feb 2018	8.8GB	Version 2 with 501m hashes and counts of password usage ordered by most to least prevalent	c267424e7d2bb5b10adff4d776fa14b0967bf0cc

<https://haveibeenpwned.com>

@samnewman

# CHECK FOR BREACHED PASSWORDS!

## Finding Pwned Passwords with 1Password

February 22, 2018 / 68 Comments / in News, Security, Watchtower / by Shiner

Yesterday, Troy Hunt launched [Pwned Passwords](#), a new service that allows you to check if your passwords have been leaked on the Internet. His database now has more than **500 million passwords** collected from various breaches. Checking your own passwords against this list is immensely valuable.

We loved Troy's new service so much that we couldn't help but create a proof of concept that integrates it with 1Password. Here's how it looks:



<https://blog.agilebits.com/2018/02/22/finding-pwned-passwords-with-1password/>

## THE THREE R'S



Justin Smith [Follow](#)  
Identity and Security Geek  
Apr 19 · 7 min read

# The Three R's of Enterprise Security: Rotate, Repave, and Repair

<https://medium.com/built-to-adapt/the-three-r-s-of-enterprise-security-rotate-repave-and-repair-f64f6d6ba29d>

@samnewman

## THE ADVANCED PERSISTENT THREAT

**“At or near the top of security concerns in the datacenter is something called an Advanced Persistent Threat (APT). An APT gains unauthorized access to a network and can stay hidden for a long period of time. Its goal is usually to steal, corrupt, or ransom data.”**

*- Justin Smith, Pivotal*



**TARGET**



In the Summer of 2015, Dutch intelligence services were the first to alert their American counterparts about the cyberintrusion of the Democratic National Committee by Cozy Bear, a hacking group believed to be tied to the Russian government. Intelligence hackers from Dutch AIVD (General Intelligence and Security Service) had penetrated the Cozy Bear computer servers as well as a security camera at the entrance of their working space, located in a university building adjacent to the Red Square in Moscow.

Over the course of a few months, they saw how the Russians penetrated several U.S. institutions, including the State Department, the White House, and the DNC. On all these occasions, the Dutch alerted the U.S. intelligence services. Dutch tv programme *Nieuwsuur* and *de Volkskrant*, a prominent newspaper in The Netherlands, jointly report on Thursday. This account is based on interviews with a dozen political, diplomatic and intelligence sources in The Netherlands and the U.S. with direct knowledge of the matter. None of them wanted to speak on the record, given the classified details of the matter.

<https://nos.nl/nieuwsuur/artikel/2213767-dutch-intelligence-first-to-alert-u-s-about-russian-hack-of-democratic-party.html>

@samnewman

# **Rotate**: Short-lived Credentials

**Rotate**: Short-lived Credentials

**Repair**: Patch Your Stuff

**Rotate**: Short-lived Credentials

**Repair**: Patch Your Stuff

**Repave**: Burn It Down!

**Rotate**: Short-lived Credentials

**Repair**: Patch Your Stuff

**Repave**: Burn It Down!

# CODESPACES R.I.P.

Data Center > Cloud

## Code Spaces goes titsup FOREVER after attacker NUKES its Amazon-hosted data

Source-sharing site to close following total cloudpocalypse



18 Jun 2014 at 20:54, Neil McAllister

    547

[http://www.theregister.co.uk/2014/06/18/code\\_spaces\\_destroyed/](http://www.theregister.co.uk/2014/06/18/code_spaces_destroyed/)

## CHECK FOR LEAKED CREDENTIALS

README.md

# Gitrob: Putting the Open Source in OSINT

Gitrob is a command line tool which can help organizations and security professionals find sensitive information lingering in publicly available files on GitHub. The tool will iterate over all public organization and member repositories and match filenames against a range of patterns for files that typically contain sensitive or dangerous information.

Looking for sensitive information in GitHub repositories is not a new thing, it has been known for a while that things such as private keys and credentials can be found with GitHub's search functionality, however Gitrob makes it easier to focus the effort on a specific organization.

<https://github.com/michenriksen/gitrob>

@samnewman

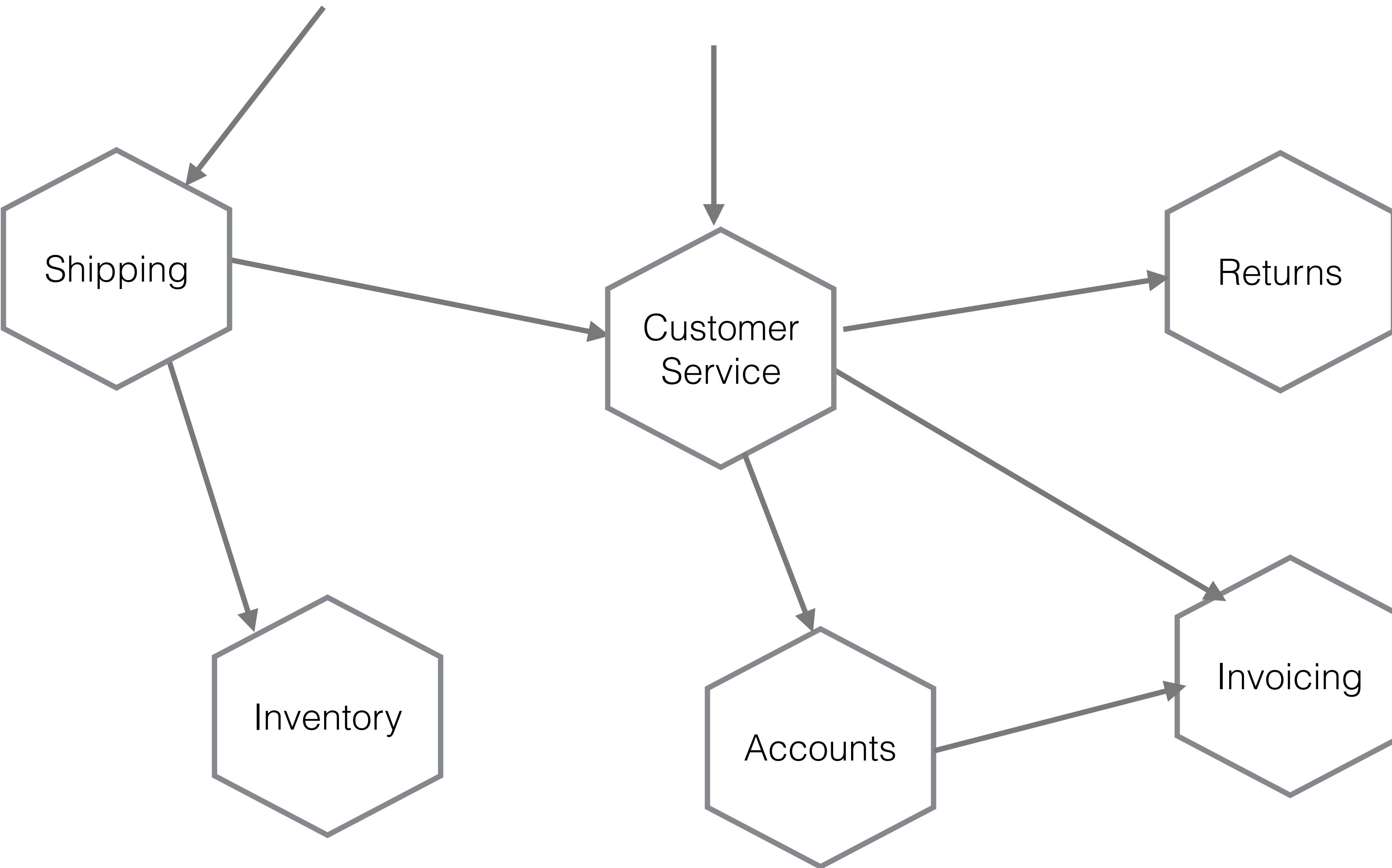
# **Revocation & Rotation Of Credentials**

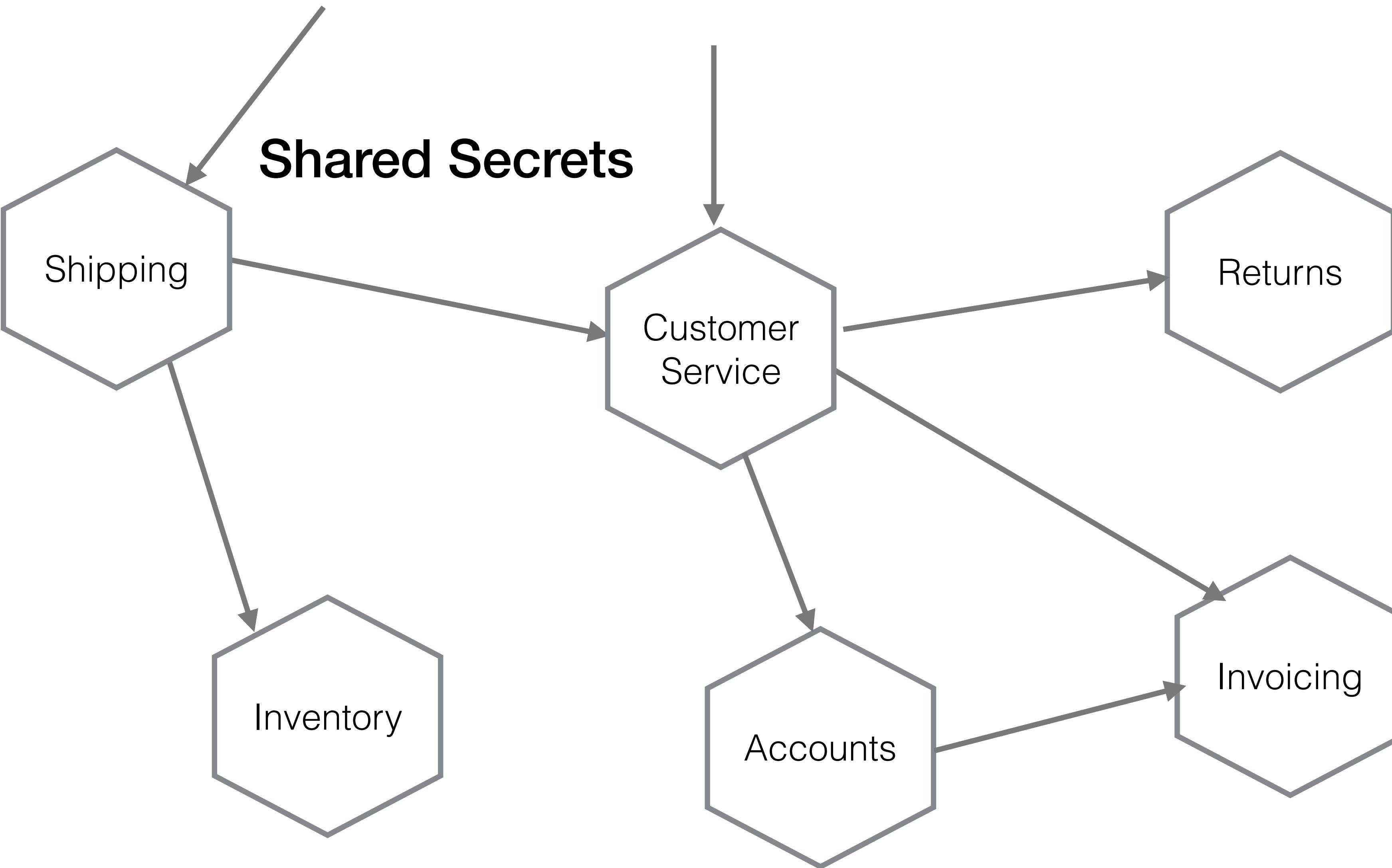
**+**

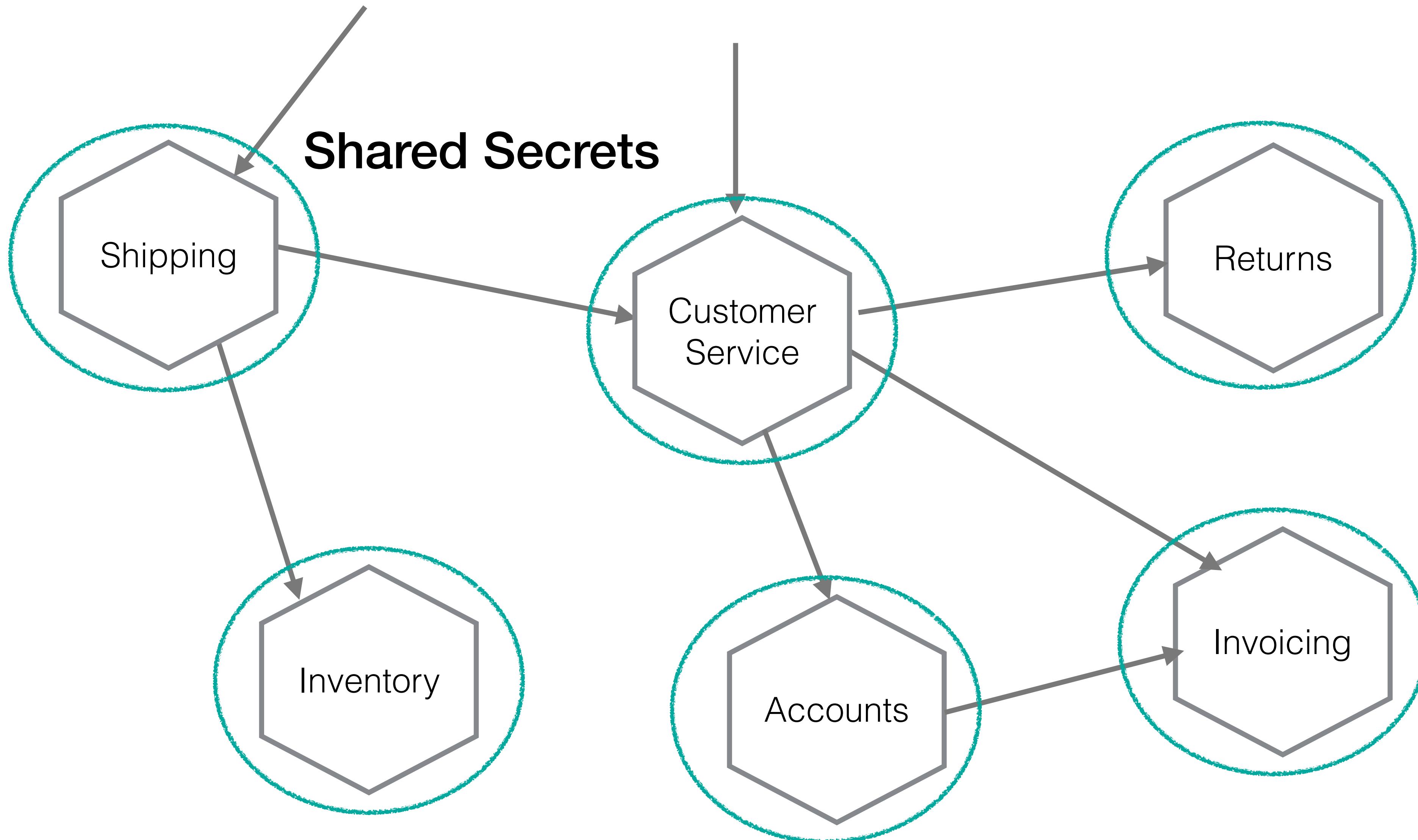
## **Microservices**

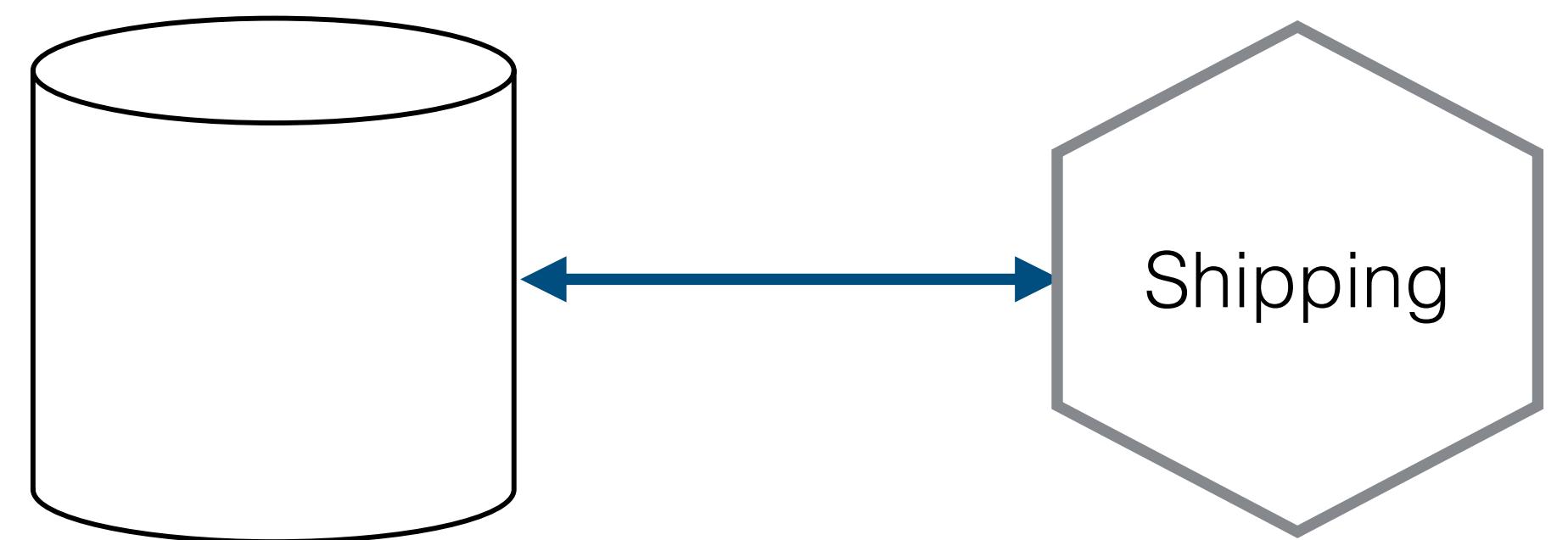
**=**

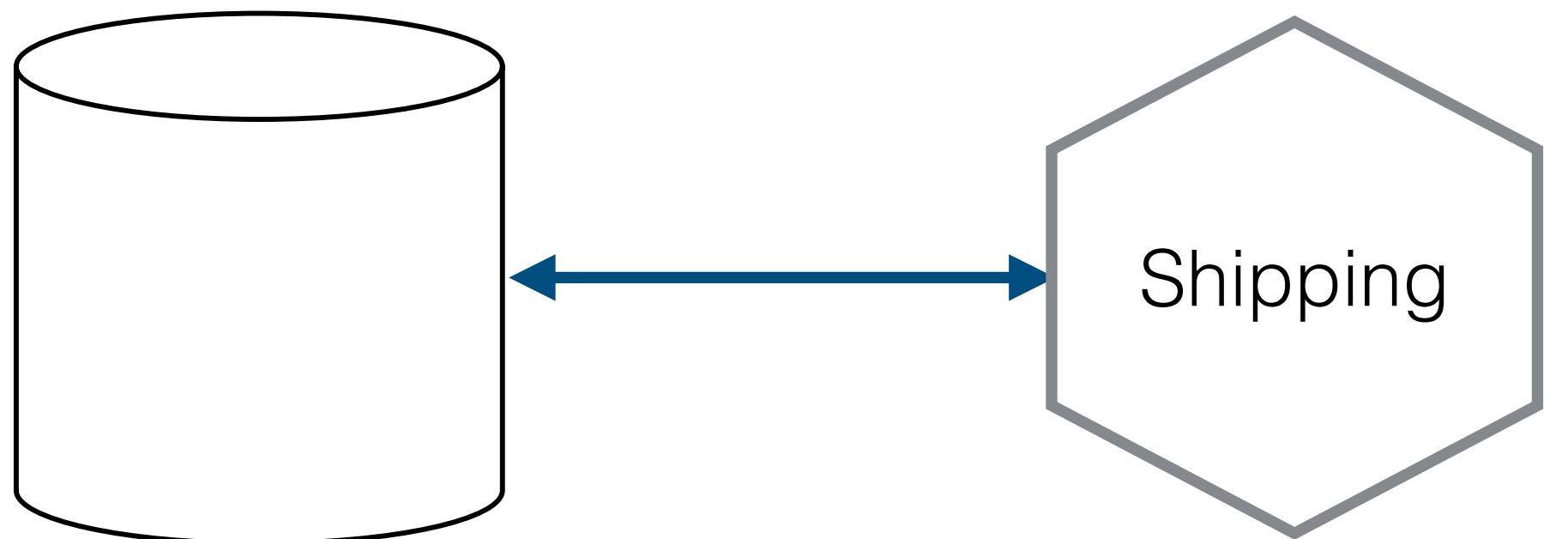
## **Pain???**



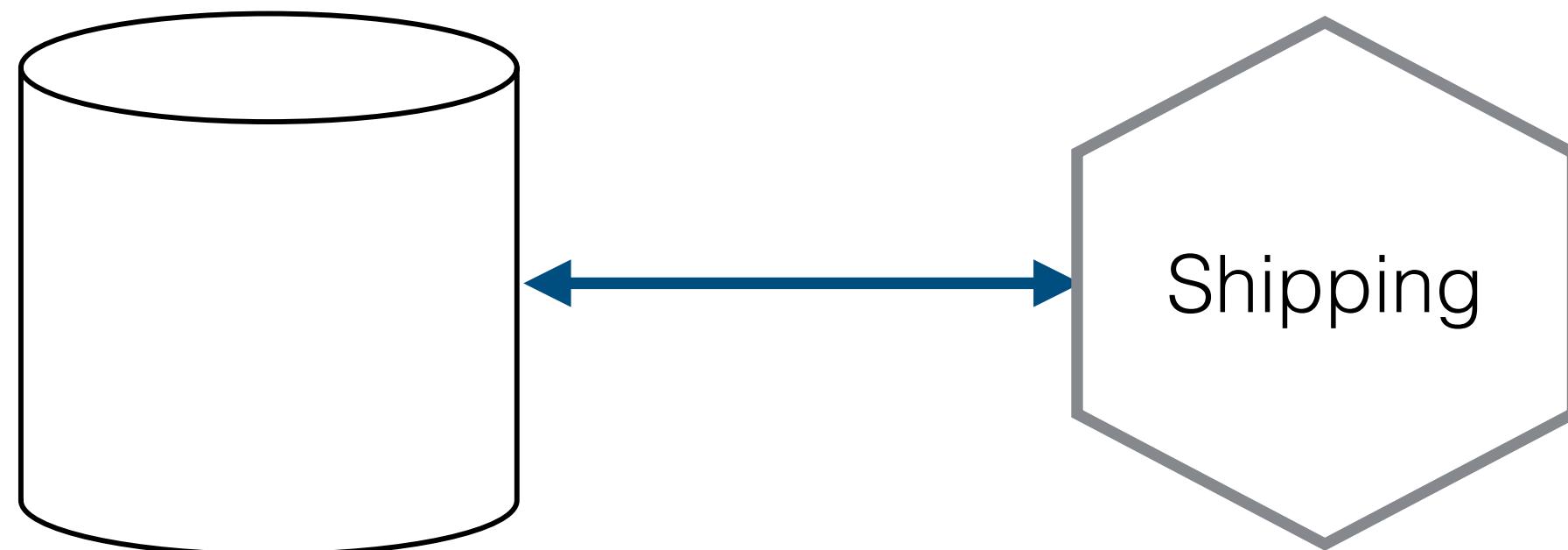






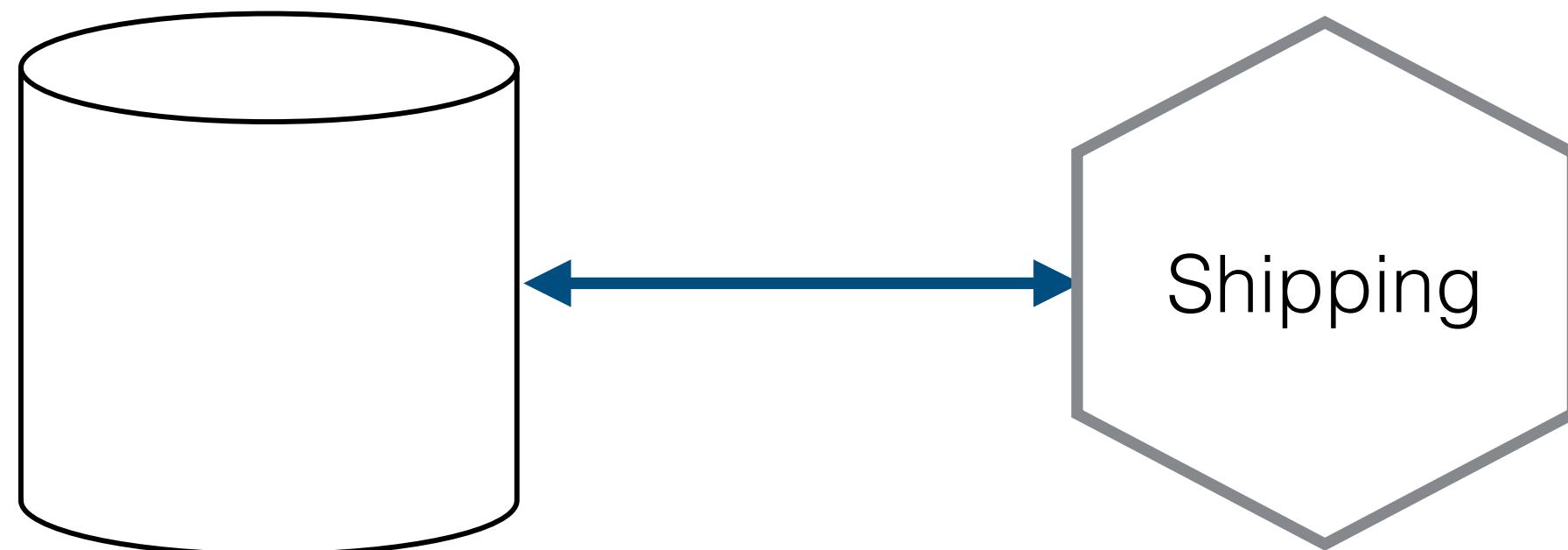


## Auth Credentials



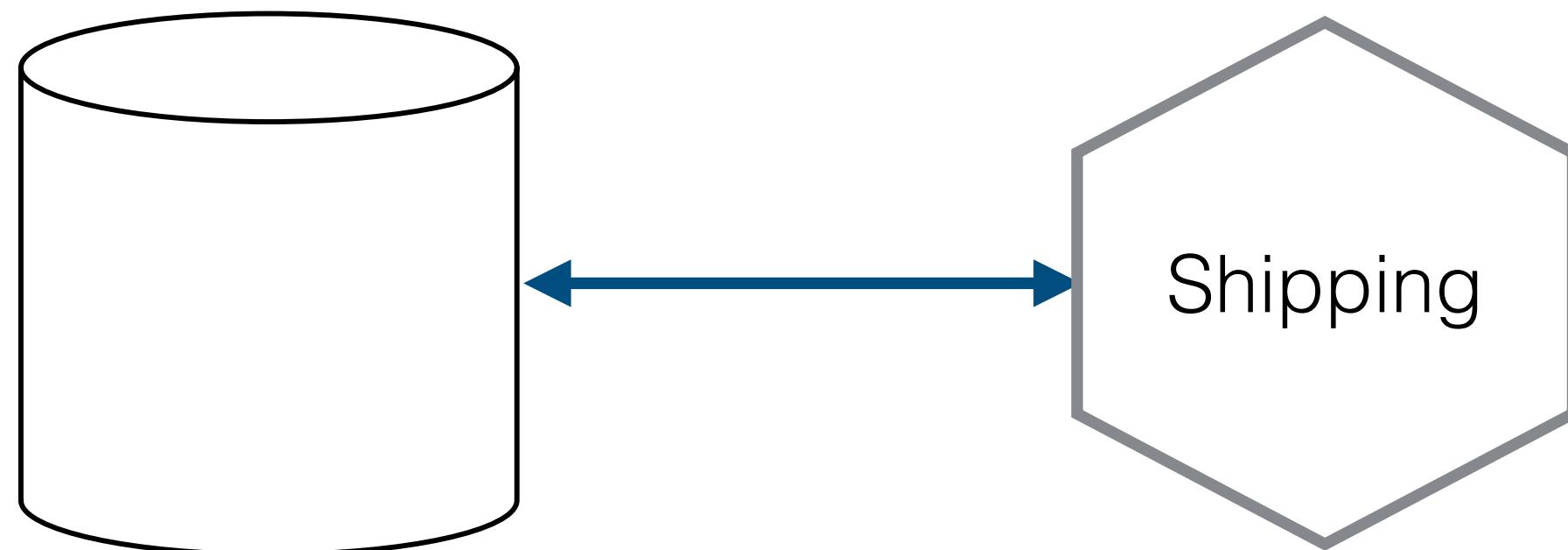
## Auth Credentials

```
DB_USERNAME = admin  
DB_PASSWORD = 123ask48321
```



## Auth Credentials

```
DB_USERNAME = admin  
DB_PASSWORD = 123ask48321
```

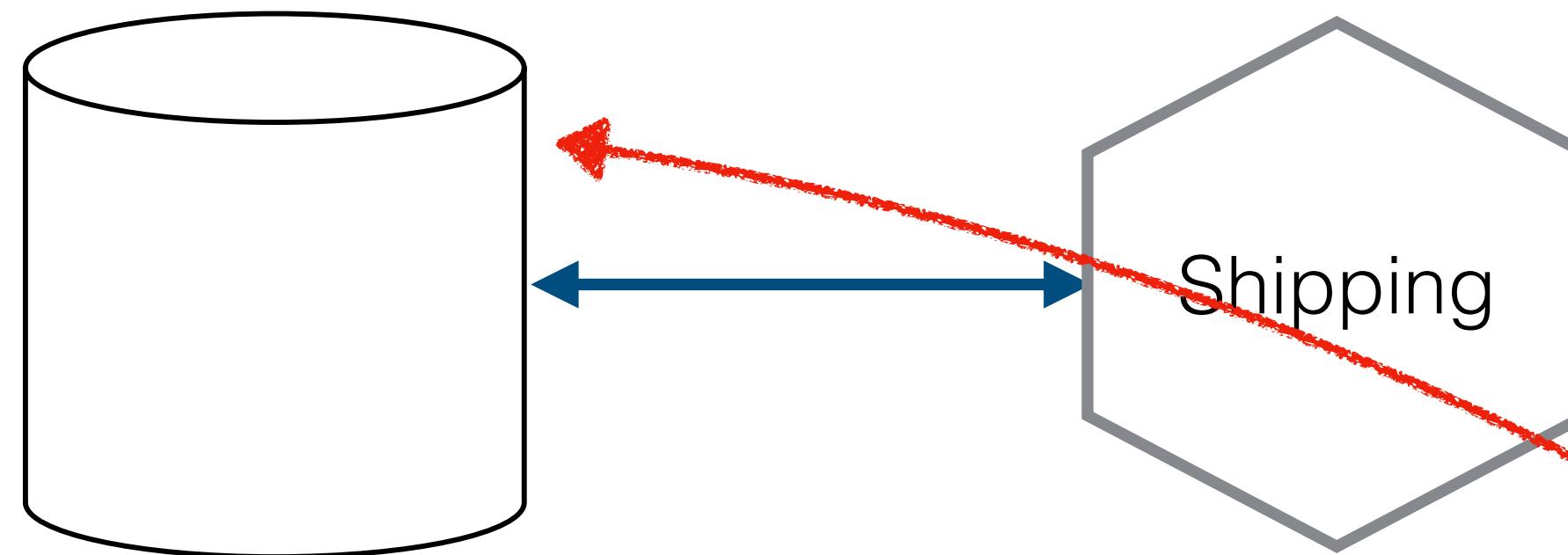


Shipping

**Auth Credentials**

```
DB_USERNAME = admin  
DB_PASSWORD = 123ask48321
```

**Leaving credentials in the open can be bad...**



**Auth Credentials**

```
DB_USERNAME = admin  
DB_PASSWORD = 123ask48321
```

**Leaving credentials in the open can be bad...**

# **Secret stores!**



A tool for managing secrets.

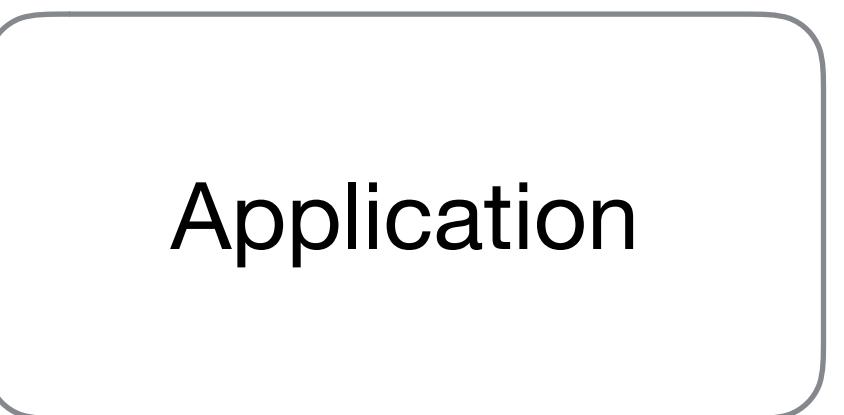
[Get Started](#)

[Launch Interactive Tutorial](#)

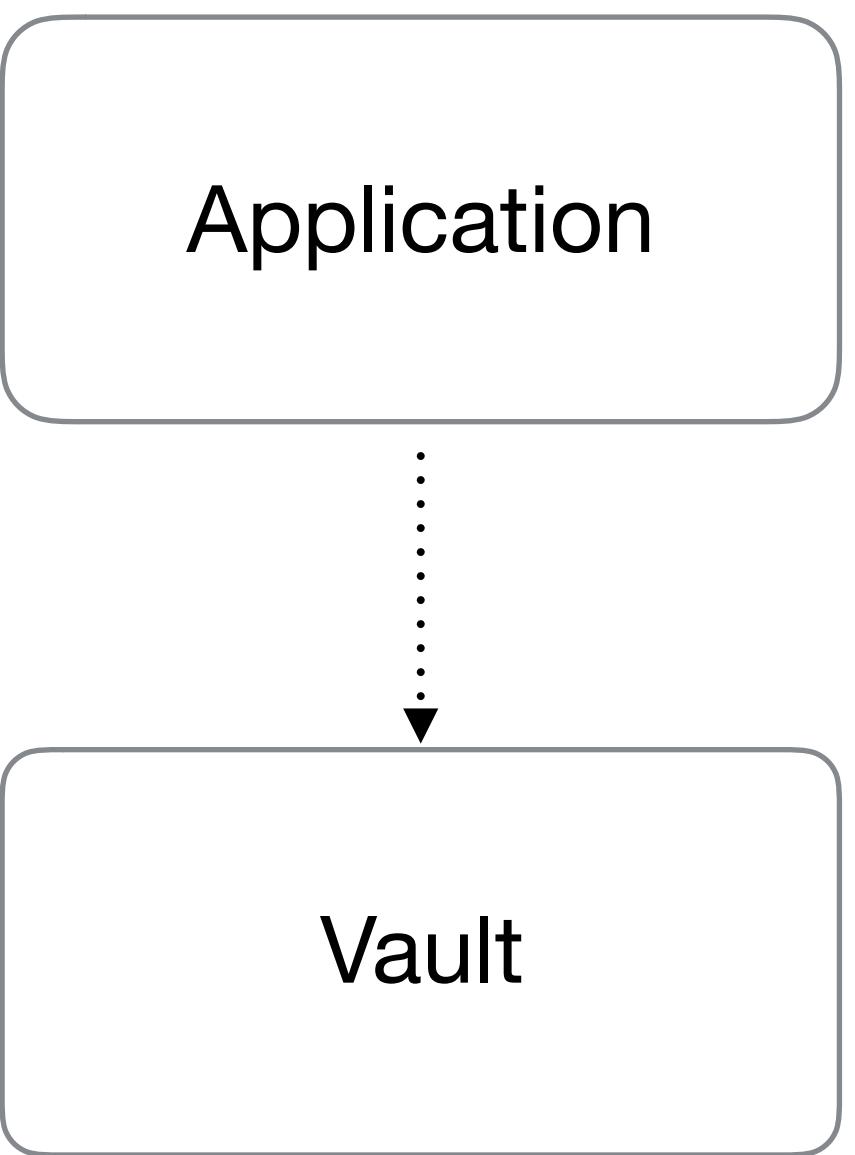




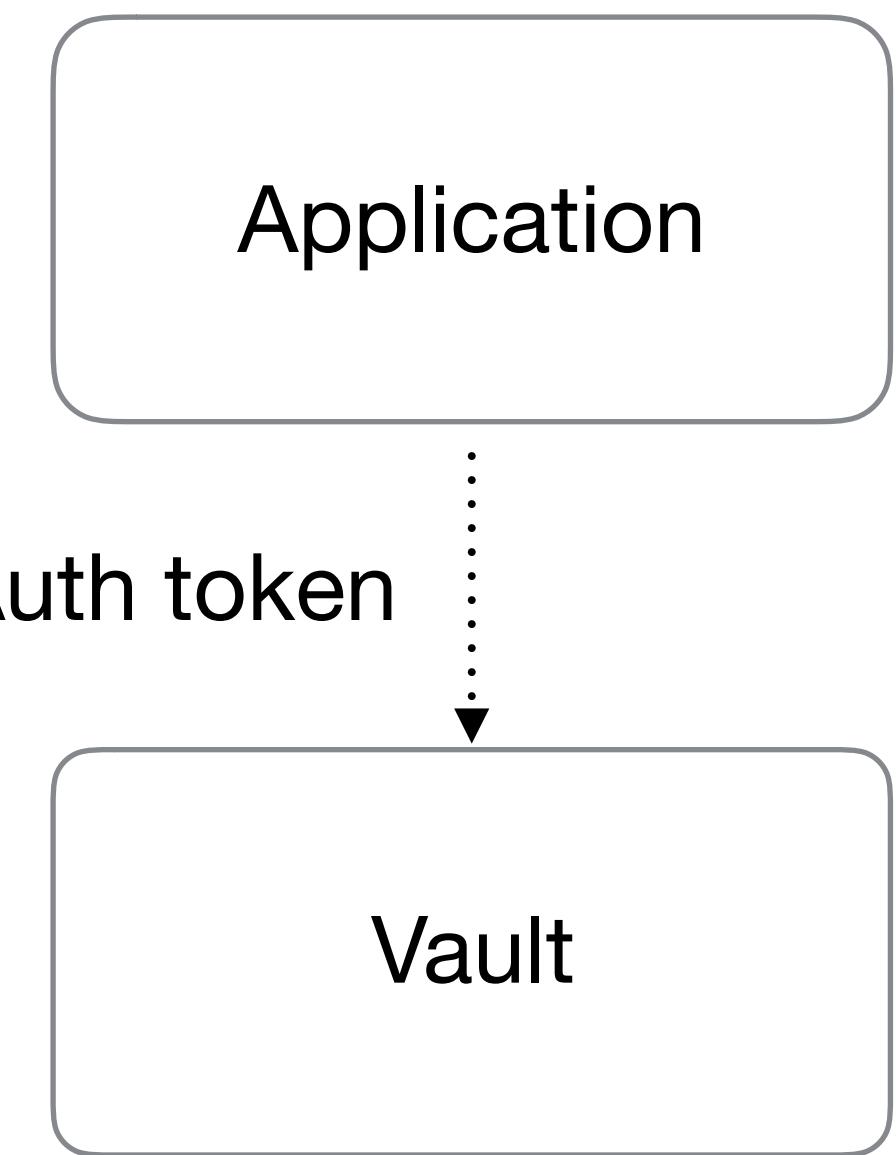
# VAULT HIGH LEVEL OVERVIEW



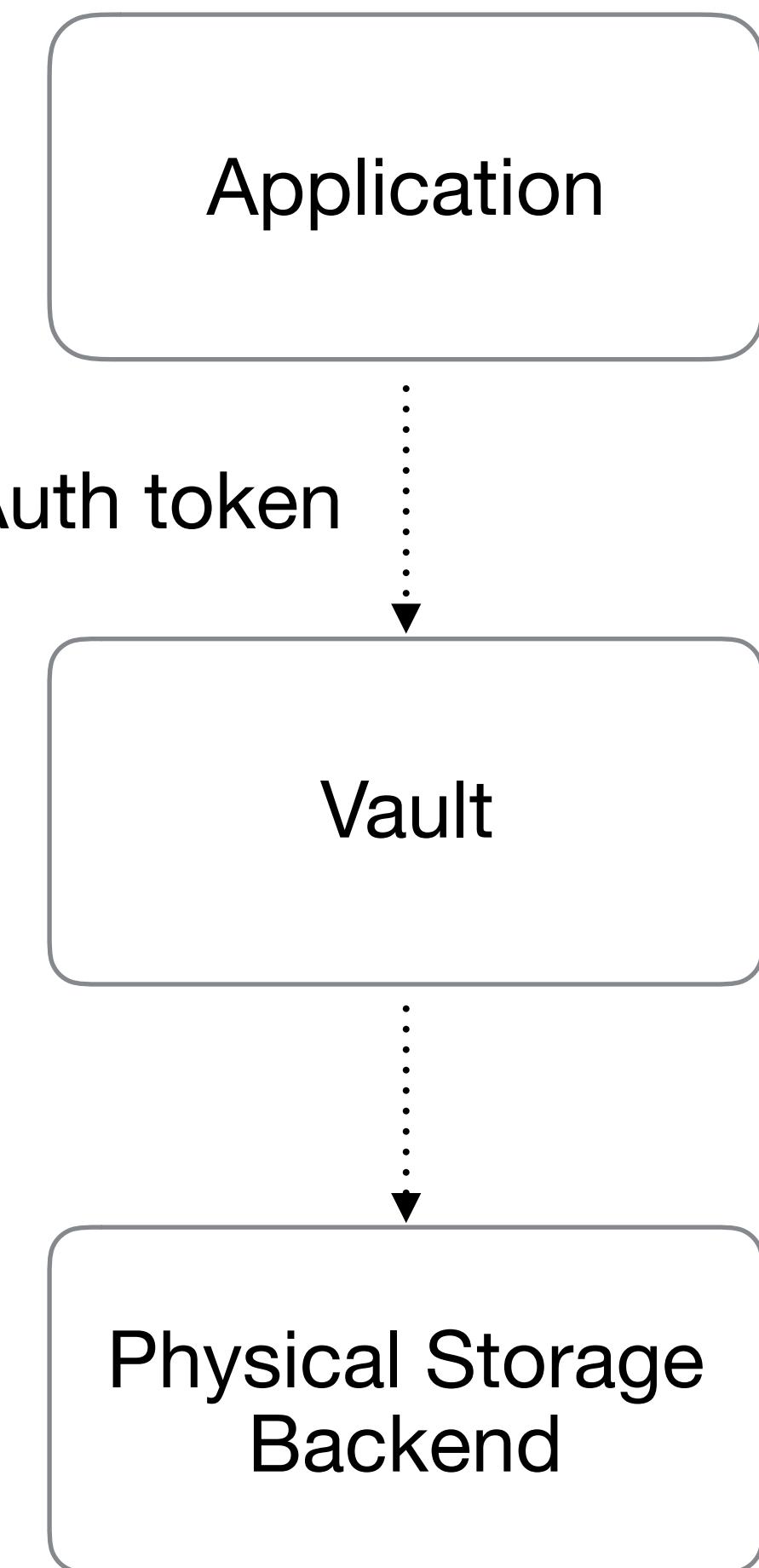
# VAULT HIGH LEVEL OVERVIEW



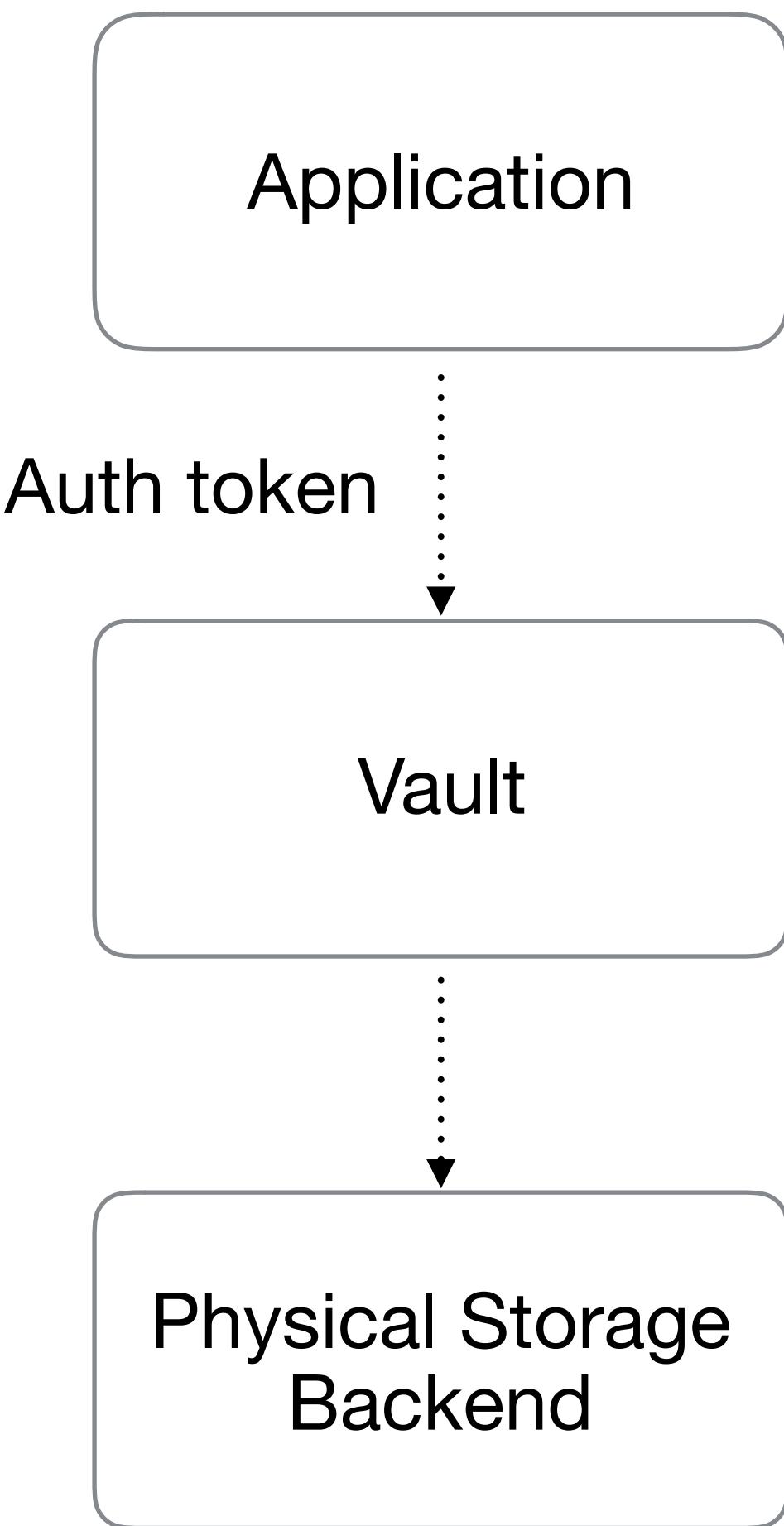
# VAULT HIGH LEVEL OVERVIEW



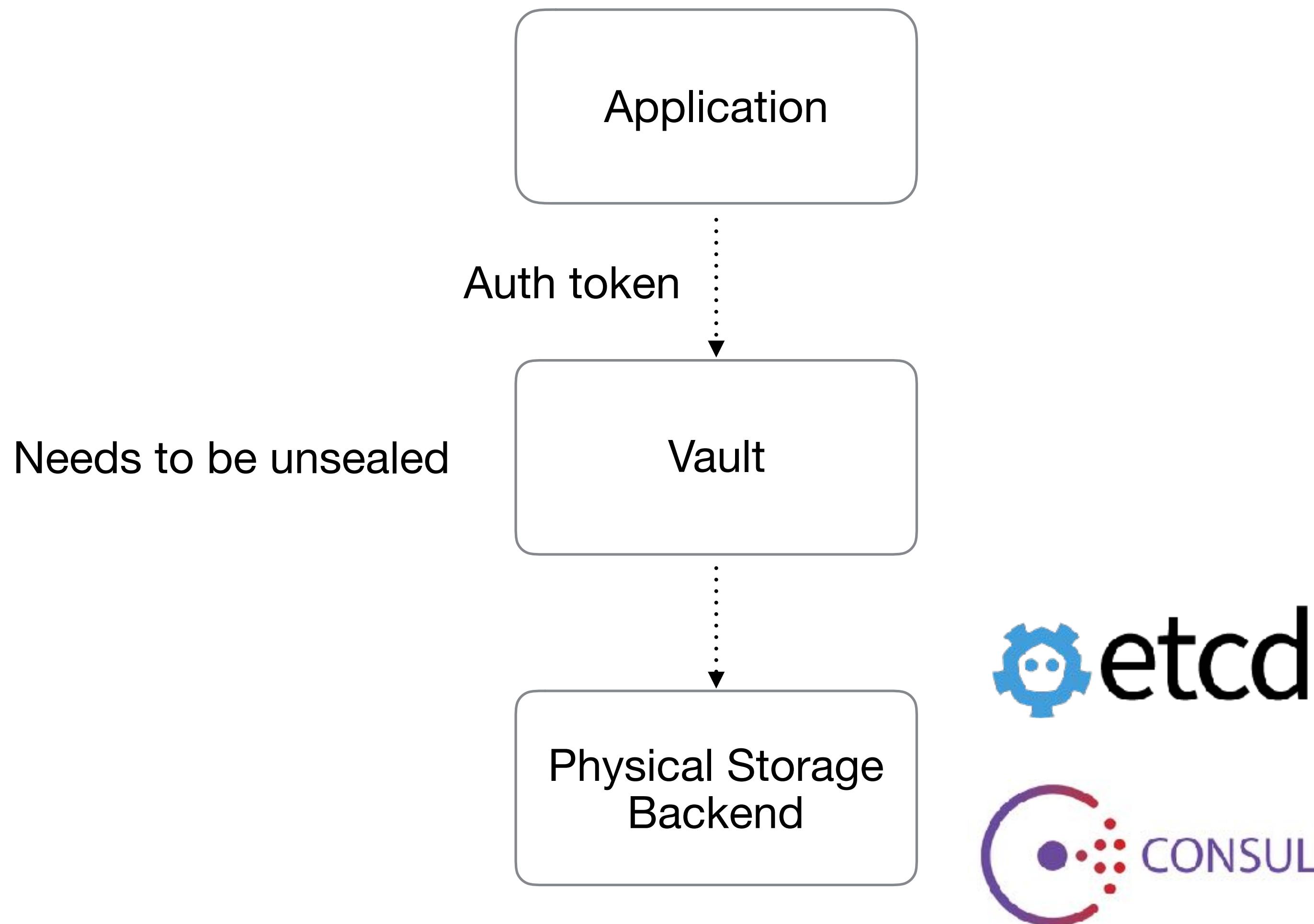
# VAULT HIGH LEVEL OVERVIEW



# VAULT HIGH LEVEL OVERVIEW



# VAULT HIGH LEVEL OVERVIEW



 etcd

 CONSUL

## WHO HAS THE KEY?



<https://www.flickr.com/photos/quinnanya/2585541255/>

# DON'T HAVE ONE KEY!



Vault



<https://www.flickr.com/photos/quinnanya/2585541255/>

@samnewman

# DON'T HAVE ONE KEY!



Vault



## Shamir's Secret Sharing

From Wikipedia, the free encyclopedia



This article may be too technical for most readers to understand. Please help improve it to make it understandable to non-experts, without removing the technical details. (March 2014) (Learn how and when to remove this template message)

**Shamir's Secret Sharing** is an algorithm in cryptography created by Adi Shamir. It is a form of [secret sharing](#), where a secret is divided into parts, giving each participant its own unique part, where some of the parts or all of them are needed in order to reconstruct the secret.

Counting on all participants to combine the secret might be impractical, and therefore sometimes the *threshold scheme* is used where any  $k$  of the parts are sufficient to reconstruct the original secret.

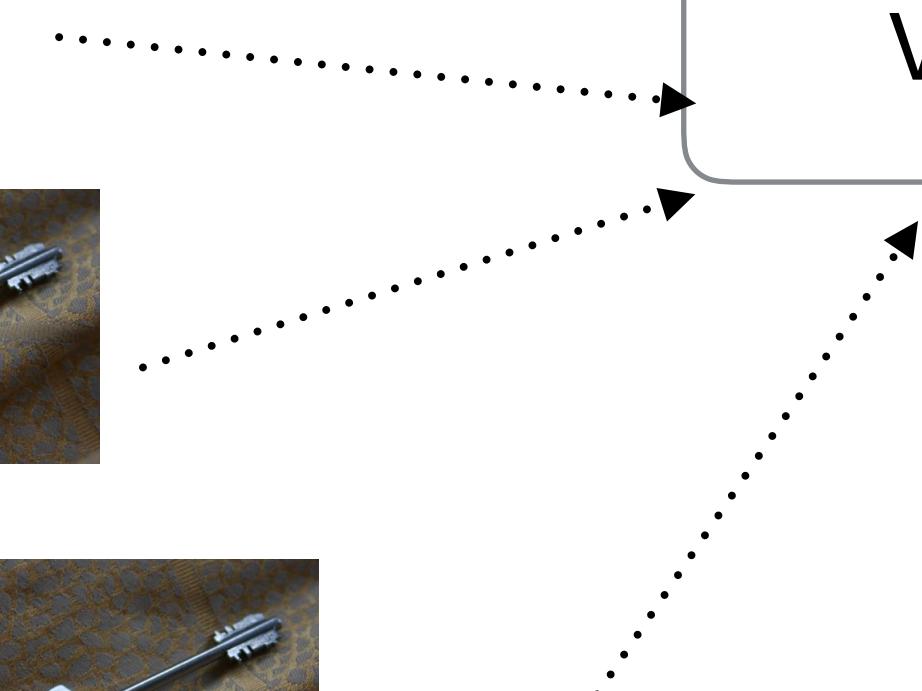
[https://en.wikipedia.org/wiki/Shamir%27s\\_Secret\\_Sharing](https://en.wikipedia.org/wiki/Shamir%27s_Secret_Sharing)

<https://www.flickr.com/photos/quinnanya/2585541255/>

# DON'T HAVE ONE KEY!



Vault



## Shamir's Secret Sharing

From Wikipedia, the free encyclopedia



This article may be too technical for most readers to understand. Please help improve it to make it understandable to non-experts, without removing the technical details. (March 2014) (Learn how and when to remove this template message)

**Shamir's Secret Sharing** is an algorithm in cryptography created by Adi Shamir. It is a form of [secret sharing](#), where a secret is divided into parts, giving each participant its own unique part, where some of the parts or all of them are needed in order to reconstruct the secret.

Counting on all participants to combine the secret might be impractical, and therefore sometimes the [threshold scheme](#) is used where any  $k$  of the parts are sufficient to reconstruct the original secret.

[https://en.wikipedia.org/wiki/Shamir%27s\\_Secret\\_Sharing](https://en.wikipedia.org/wiki/Shamir%27s_Secret_Sharing)

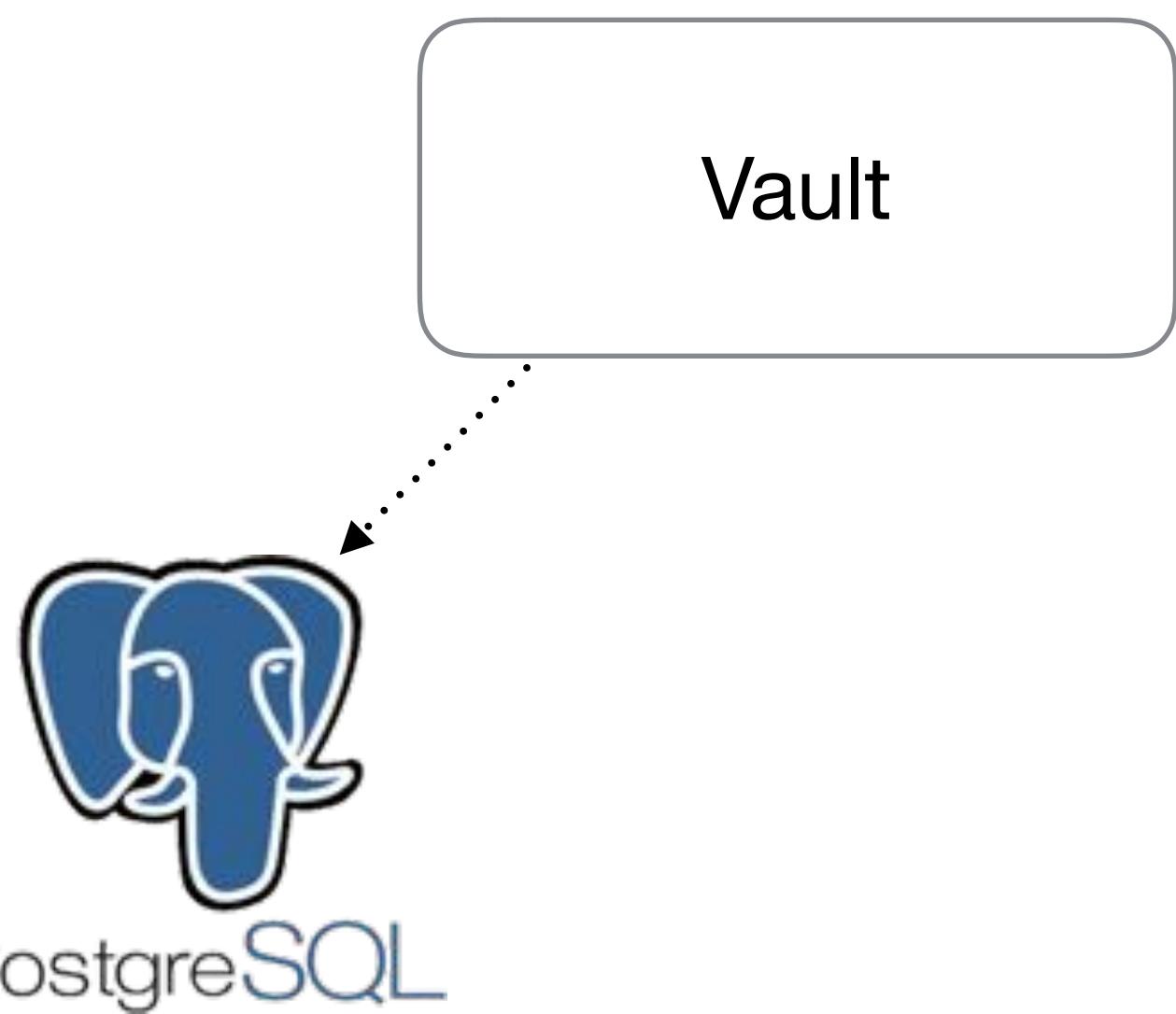
<https://www.flickr.com/photos/quinnanya/2585541255/>

@samnewman

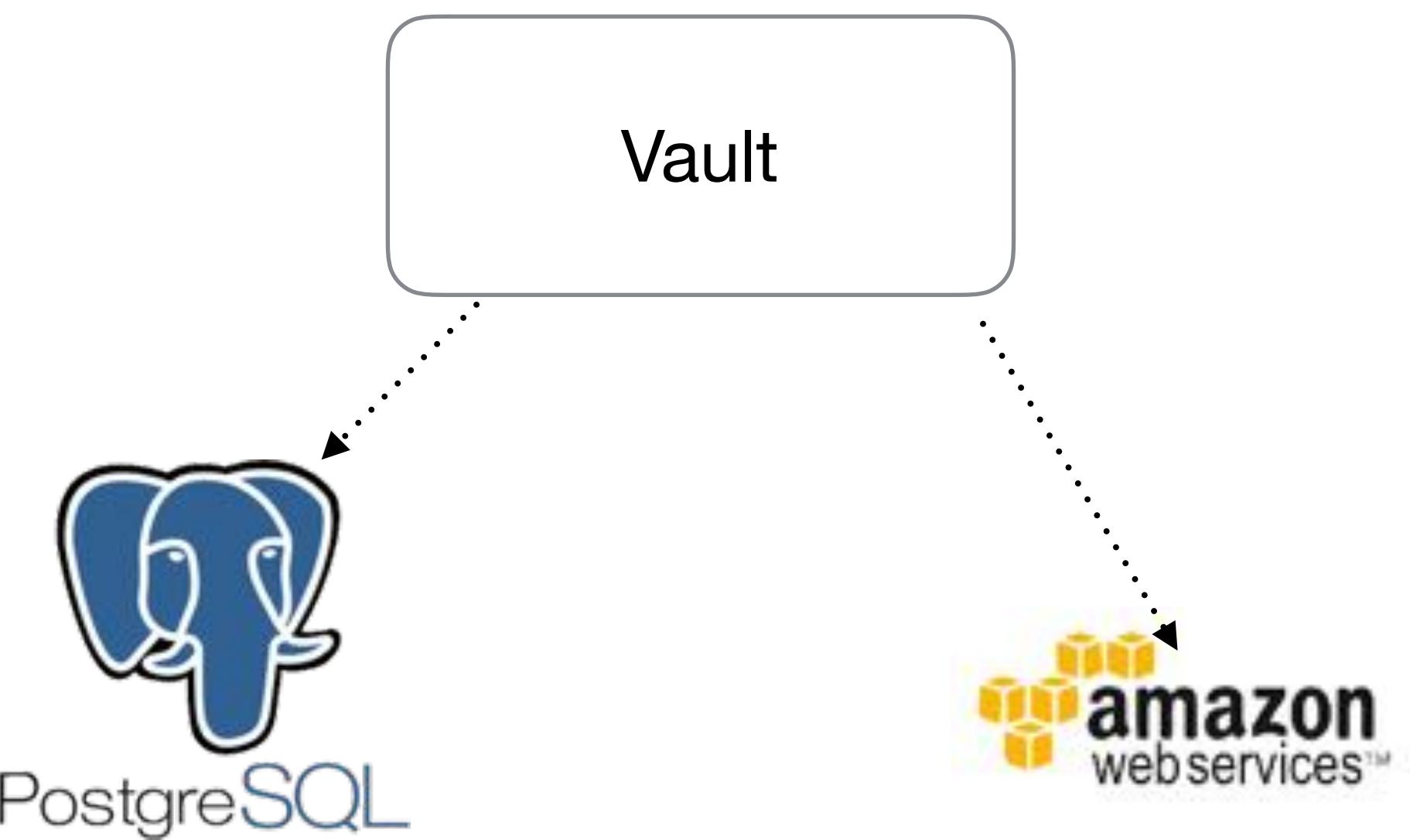
# TIME-LIMITED CREDENTIALS

Vault

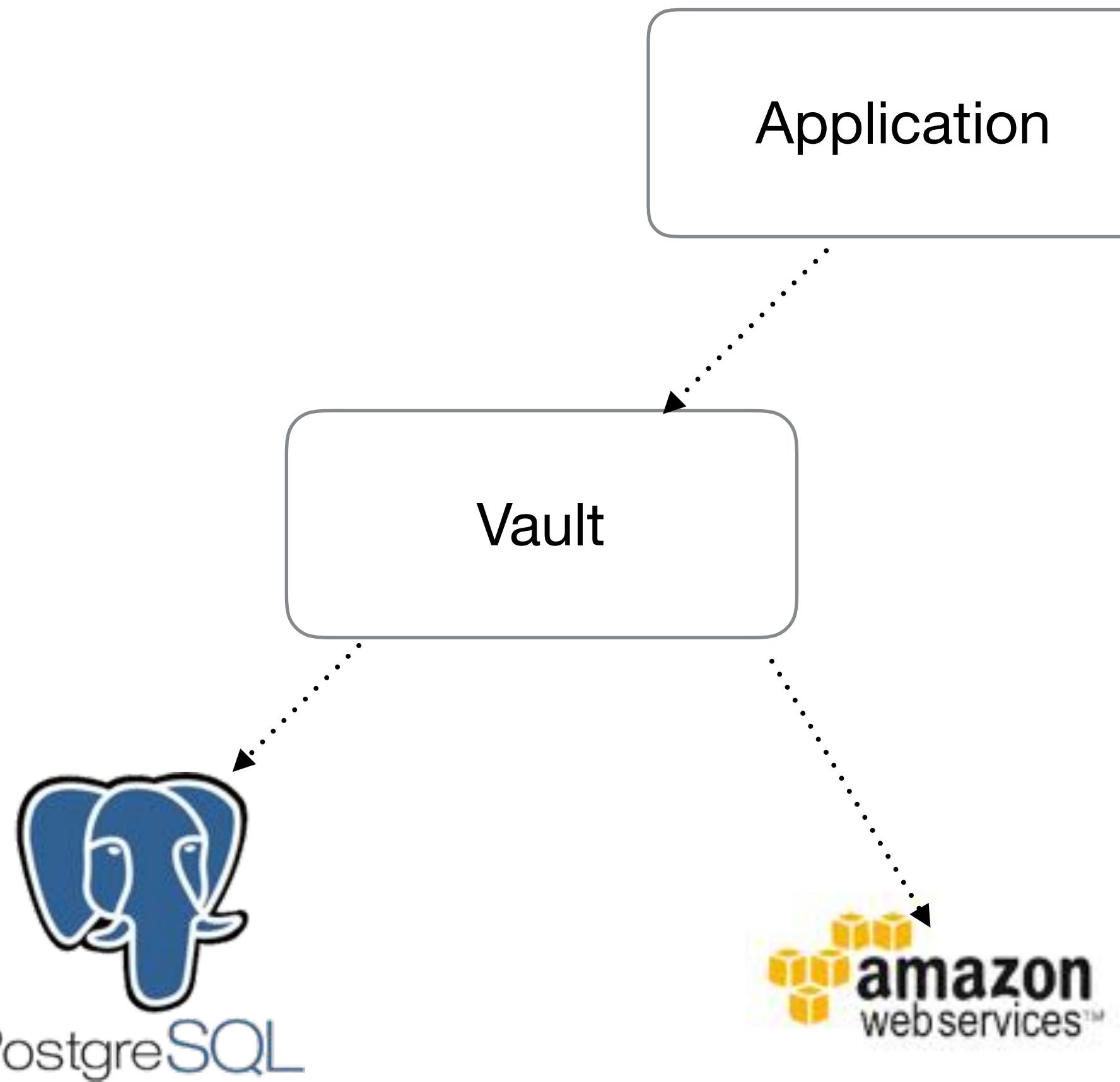
## TIME-LIMITED CREDENTIALS



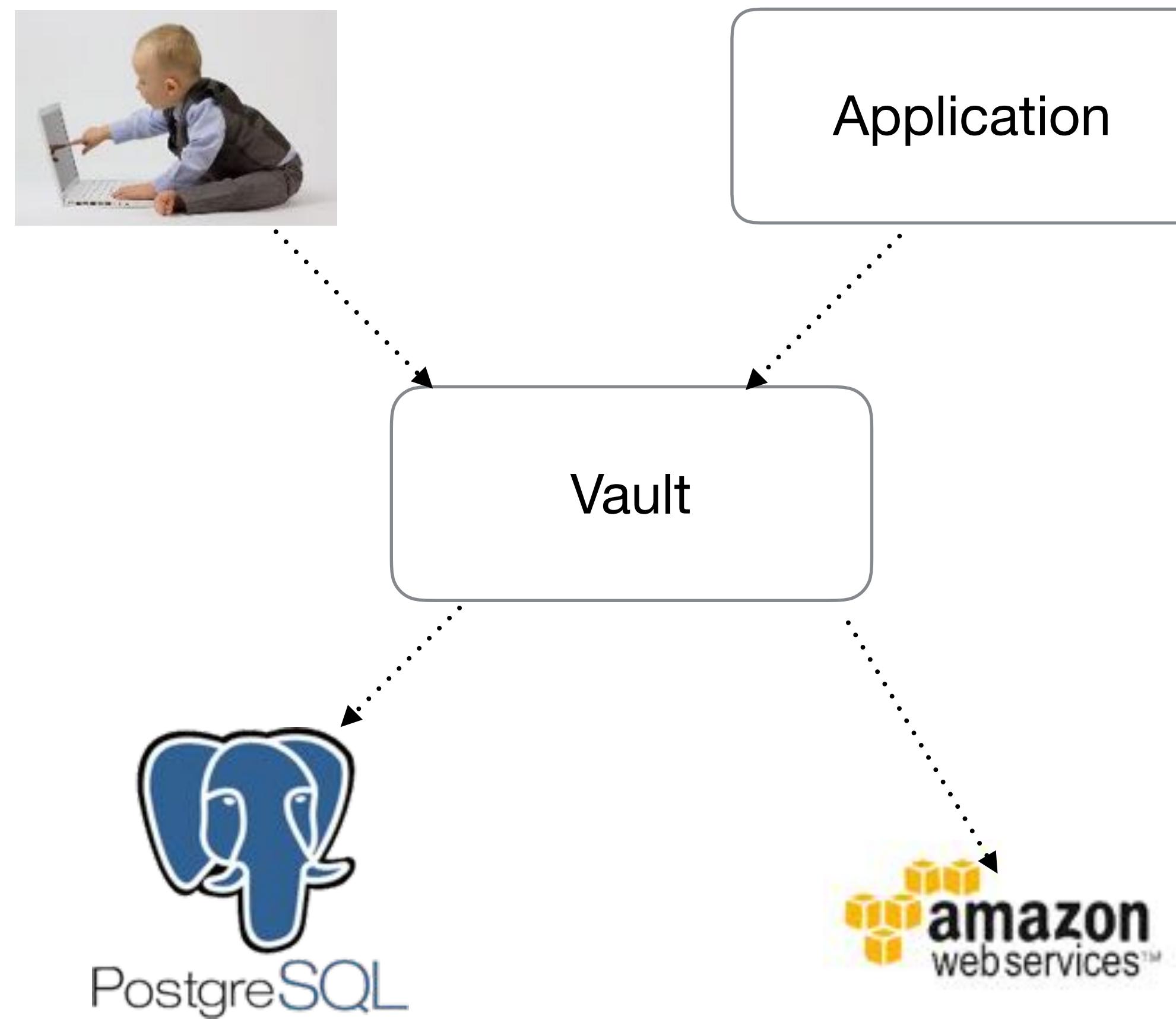
# TIME-LIMITED CREDENTIALS



# TIME-LIMITED CREDENTIALS



# TIME-LIMITED CREDENTIALS



# AWESOMENESS



<https://github.com/hashicorp/consul-template>

# AWESOMENESS



CONSUL  
TEMPLATE

<https://github.com/hashicorp/consul-template>

```
adapter: postgresql
host: {{key "my-app/production/host"}}
username: {{$secret.Data.username}}
password: {{$secret.Data.password}}
{{end}}
```

From: <https://www.hashicorp.com/blog/using-vault-with-consul-template>

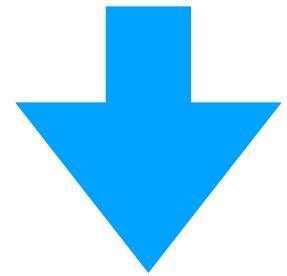
# AWESOMENESS



CONSUL  
TEMPLATE

<https://github.com/hashicorp/consul-template>

```
adapter: postgresql
host: {{key "my-app/production/host"}}
username: {{$secret.Data.username}}
password: {{$secret.Data.password}}
{{end}}
```



From: <https://www.hashicorp.com/blog/using-vault-with-consul-template>

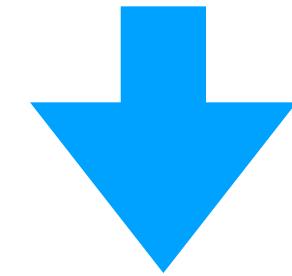
# AWESOMENESS



CONSUL  
TEMPLATE

<https://github.com/hashicorp/consul-template>

```
adapter: postgresql
host: {{key "my-app/production/host"}}
username: {{$secret.Data.username}}
password: {{$secret.Data.password}}
{{end}}
```



```
adapter: postgresql
host: db-service-183.corp.com
username: as15593kd235423
password: fklk11492309482
{{end}}
```

From: <https://www.hashicorp.com/blog/using-vault-with-consul-template>

## WHAT ELSE CAUSES BREACHES?

**“44 percent of security breaches occur after vulnerabilities and solutions have been identified. In other words, the problems could have been avoided if found vulnerabilities had been addressed sooner.”**

**- Forbes/BMC, 2016**

## Massive Equifax data breach - what you need to know



By **Callum Mason**, News Reporter  
12 Sep 2017 | Updated 18 Sep 2017



Credit report heavyweight Equifax has warned that up to 400,000 UK consumers may have had their personal details stolen as part of a massive global data breach. Info on exactly who's been affected and what you can do about it is still somewhat sketchy, but here's what we know.

Equifax revealed on 8 September that 143 million consumers in the US could have been affected by the incident, which saw hackers access data such as names, address and dates of birth, as well as credit card numbers in a smaller number of cases.

Although its UK business – Equifax Ltd – now says systems in this country are not affected, it admits a file which was stored in the US and contained more limited personal information on up to 400,000 UK consumers may have been accessed.

### Related MSE Guides

[Credit Scores](#)  
Bust myths & improve your score

[30+ Ways to Stop Scams](#)  
As scams get clever, we need to too!

[Check your credit report for free](#)  
Grab your file and check your score, or even get PAID to do it



### Get Our Free Money Tips Email!

For all the latest deals, guides and loopholes - join the 12m who get it.  
*Don't miss out*

Enter Email Address

GET IT!

[FAQs](#) | [Privacy Policy](#) | [Past Emails](#) | [Unsubscribe](#)

### What is Equifax and what data does it have?

Equifax is the second biggest credit reference agency in the UK, after Experian.

<https://www.moneysavingexpert.com/news/protect/2017/09/massive-equifax-data-breach---what-you-need-to-know>

# PATCH MUCH?

## Equifax confirms march struts vulnerability behind breach

by Chris Brook for Threat Post

Equifax said the culprit September 14, 2017, 4:00 pm  
behind this summer's massive breach of 143 million Americans was indeed CVE-2017-5638,  
an Apache Struts vulnerability patched back in March.

The bug was widely assumed by experts to be the "U.S. website application vulnerability" implicated by the company last Thursday, especially after an Apache spokeswoman told Reuters on Friday that it appeared the consumer credit reporting agency hadn't applied patches for flaws discovered earlier this year.

On Wednesday company specified the flaw in a statement posted to its site and stressed it was continuing to work alongside law enforcement to investigate the incident.

<https://www.pinkconnect.com/equifax-confirms-march-struts-vulnerability-behind-breach/>

@samnewman

# PATCH MUCH?

## Equifax confirms march struts vulnerability behind breach

by Chris Brook for Threat Post

Equifax said the culprit behind this summer's massive breach of 143 million Americans was indeed CVE-2017-5638, an Apache Struts vulnerability patched back in March.

The bug was widely assumed by experts to be the "U.S. website application vulnerability" implicated by the company last Thursday. Reuters on Friday that it appeared the company had applied patches for flaws discovered earlier this month.

On Wednesday company specified the flaw was continuing to work alongside law enforcement.

"Equifax has been intensely investigating the scope of the intrusion with the assistance of a leading, independent cybersecurity firm to determine what information was accessed and who has been impacted. We know that criminals exploited a U.S. website application vulnerability. The vulnerability was Apache Struts CVE-2017-5638. We continue to work with law enforcement as part of our criminal investigation, and have shared indicators of compromise with law enforcement."

<https://www.pinkconnect.com/equifax-confirms-march-struts-vulnerability-behind-breach/>

@samnewman

# PATCH MUCH?

## Equifax confirms march struts vulnerability behind breach

by Chris Brook for Threat Post Equifax said the culprit

September 14, 2017, 4:00 pm

behind this summer's massive breach of 143 million Americans was indeed CVE-2017-5638, an Apache Struts vulnerability patched back in March.

The bug was widely assumed by experts to be the "U.S. website application vulnerability" implicated by the company last Thursday. Reuters on Friday that it appeared the company had applied patches for flaws discovered earlier this year.

On Wednesday company specified the flaw was continuing to work alongside law enforcement.

# CVE-2017-5638

"Equifax has been intensely investigating the scope of the intrusion with the assistance of a leading, independent cybersecurity firm to determine what information was accessed and who has been impacted. We know that criminals exploited a U.S. website application vulnerability. The vulnerability was Apache Struts CVE-2017-5638. We continue to work with law enforcement as part of our criminal investigation, and have shared indicators of compromise with law enforcement."

<https://www.pinkconnect.com/equifax-confirms-march-struts-vulnerability-behind-breach/>

@samnewman

# CVE-2017-5638

## Current Description

The Jakarta Multipart parser in Apache Struts 2 2.3.x before 2.3.32 and 2.5.x before 2.5.10.1 has incorrect exception handling and error message generation during file-upload attempts, which allows remote attackers to execute arbitrary commands via a crafted Content-Type, Content-Disposition, or Content-Length HTTP header, as exploited in the wild in March 2017 with a Content-Type header containing a #cmd= string.

Source: MITRE Last Modified: 09/22/2017 [+View Analysis Description](#)

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (legend)

Impact Score: 6.0

Exploitability Score: 3.9

# CVE-2017-5638

## Current Description

The Jakarta Multipart parser in Apache Struts 2 2.3.x before 2.3.32 and 2.5.x before 2.5.10.1 has incorrect exception handling and error message generation during file-upload attempts, which allows remote attackers to execute arbitrary commands via a crafted Content-Type, Content-Disposition, or Content-Length HTTP header, as exploited in the wild in March 2017 with a Content-Type header containing a #cmd= string.

Source: MITRE Last Modified: 09/22/2017 [+View Analysis Description](#)

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (legend)

Impact Score: 6.0

Exploitability Score: 3.9

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

# CVE-2017-5638

## Current Description

The Jakarta Multipart parser in Apache Struts 2 2.3.x before 2.3.32 and 2.5.x before 2.5.10.1 has incorrect exception handling and error message generation during file-upload attempts, which allows remote attackers to execute arbitrary commands via a crafted Content-Type, Content-Disposition, or Content-Length HTTP header, as exploited in the wild in March 2017 with a Content-Type header containing a #cmd= string.

Source: MITRE Last Modified: 09/22/2017 [+View Analysis Description](#)

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (legend)

Impact Score: 6.0

Exploitability Score: 3.9

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

# CVE-2017-5638

## Current Description

The Jakarta Multipart parser in Apache Struts 2 2.3.x before 2.3.32 and 2.5.x before 2.5.10.1 has incorrect exception handling and error message generation during file-upload attempts, which allows remote attackers to execute arbitrary commands via a crafted Content-Type, Content-Disposition, or Content-Length HTTP header, as exploited in the wild in March 2017 with a Content-Type header containing a #cmd= string.

Source: MITRE Last Modified: 09/22/2017 [+View Analysis Description](#)

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (legend)

Impact Score: 6.0

Exploitability Score: 3.9

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Reported March 2017

# CVE-2017-5638

## Current Description

The Jakarta Multipart parser in Apache Struts 2 2.3.x before 2.3.32 and 2.5.x before 2.5.10.1 has incorrect exception handling and error message generation during file-upload attempts, which allows remote attackers to execute arbitrary commands via a crafted Content-Type, Content-Disposition, or Content-Length HTTP header, as exploited in the wild in March 2017 with a Content-Type header containing a #cmd= string.

Source: MITRE Last Modified: 09/22/2017 [+View Analysis Description](#)

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (legend)

Impact Score: 6.0

Exploitability Score: 3.9

## Impact

CVSS Severity (version 3.0):

CVSS v3 Base Score: 10.0 Critical

Reported March 2017

Patched in struts 2.3.32 / 2.5.10.1 on 7th March

# EQUIFAX TIMELINE

sources: <https://krebsonsecurity.com/2017/09/breach-at-equifax-may-impact-143m-americans/>  
<https://www.usatoday.com/story/tech/2017/09/26/timeline-events-surrounding-equifax-data-breach/703691001/>

@samnewman

## EQUIFAX TIMELINE

Equifax breach happened between mid-May and July

sources: <https://krebsonsecurity.com/2017/09/breach-at-equifax-may-impact-143m-americans/>  
<https://www.usatoday.com/story/tech/2017/09/26/timeline-events-surrounding-equifax-data-breach/703691001/>

@samnewman

## EQUIFAX TIMELINE

Equifax breach happened between mid-May and July

Equifax spotted it on July 29th

sources: <https://krebsonsecurity.com/2017/09/breach-at-equifax-may-impact-143m-americans/>  
<https://www.usatoday.com/story/tech/2017/09/26/timeline-events-surrounding-equifax-data-breach/703691001/>

@samnewman

## EQUIFAX TIMELINE

Equifax breach happened between mid-May and July

Equifax spotted it on July 29th

Reported on September 7th

sources: <https://krebsonsecurity.com/2017/09/breach-at-equifax-may-impact-143m-americans/>  
<https://www.usatoday.com/story/tech/2017/09/26/timeline-events-surrounding-equifax-data-breach/703691001/>

@samnewman

## EQUIFAX TIMELINE

Equifax breach happened between mid-May and July

Equifax spotted it on July 29th

Reported on September 7th

At the time the breach was discovered, the patch had been out for at least 2 months, and perhaps as long as 4 months

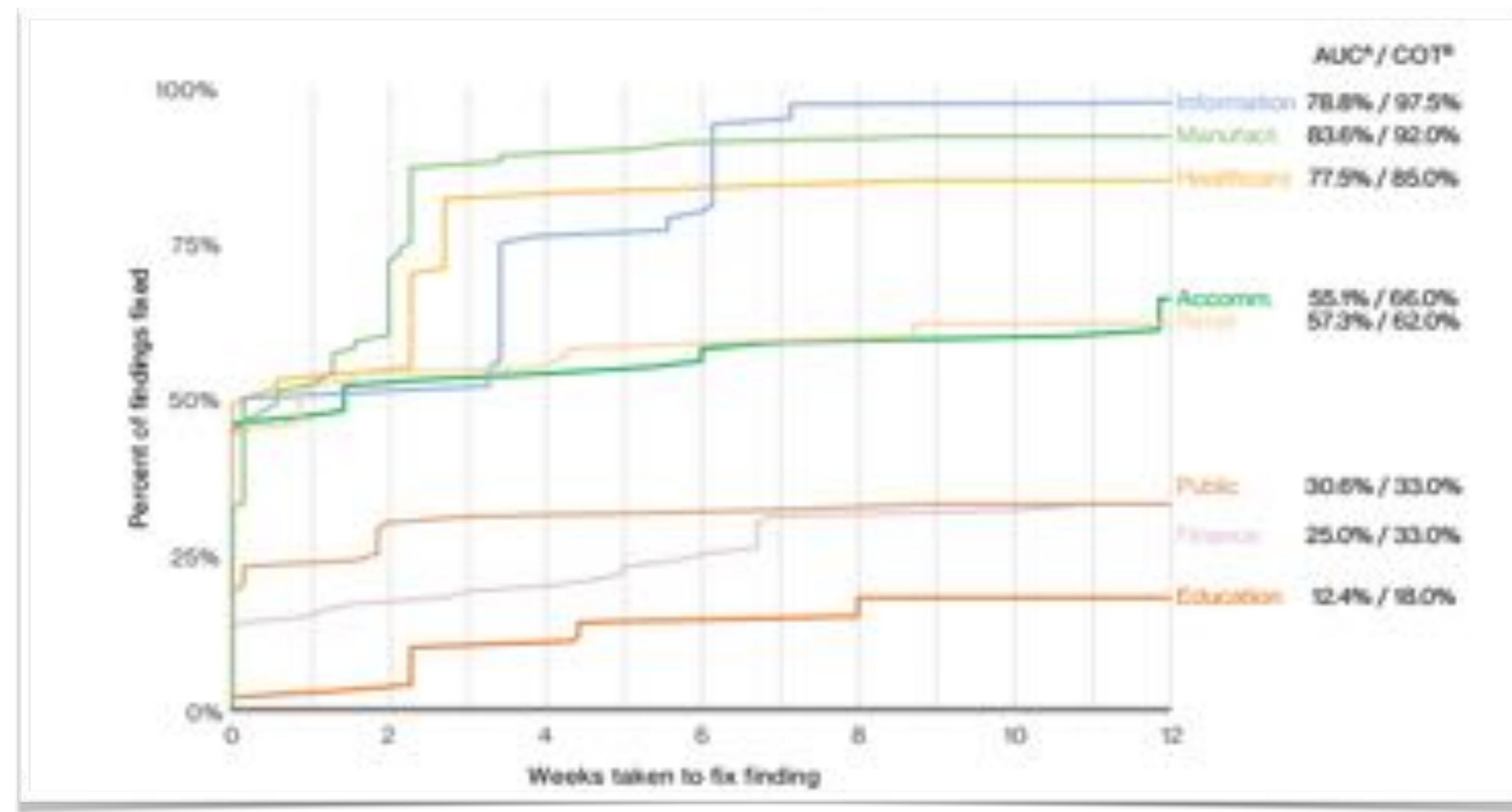
sources: <https://krebsonsecurity.com/2017/09/breach-at-equifax-may-impact-143m-americans/>  
<https://www.usatoday.com/story/tech/2017/09/26/timeline-events-surrounding-equifax-data-breach/703691001/>

@samnewman

**2 to 4 months**

**Hands up if you \*know\* you update your  
3rd party libraries for all your code every  
2-4 months?**

## PATCHING HYGIENE





# PATCHING MADNESS!

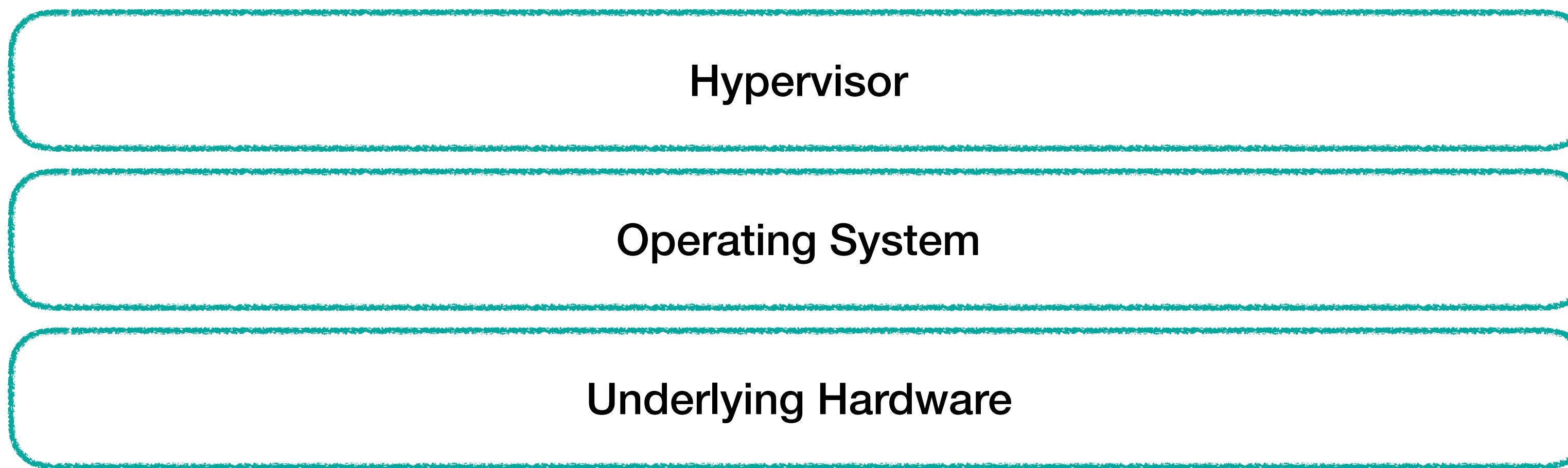
Underlying Hardware

# PATCHING MADNESS!

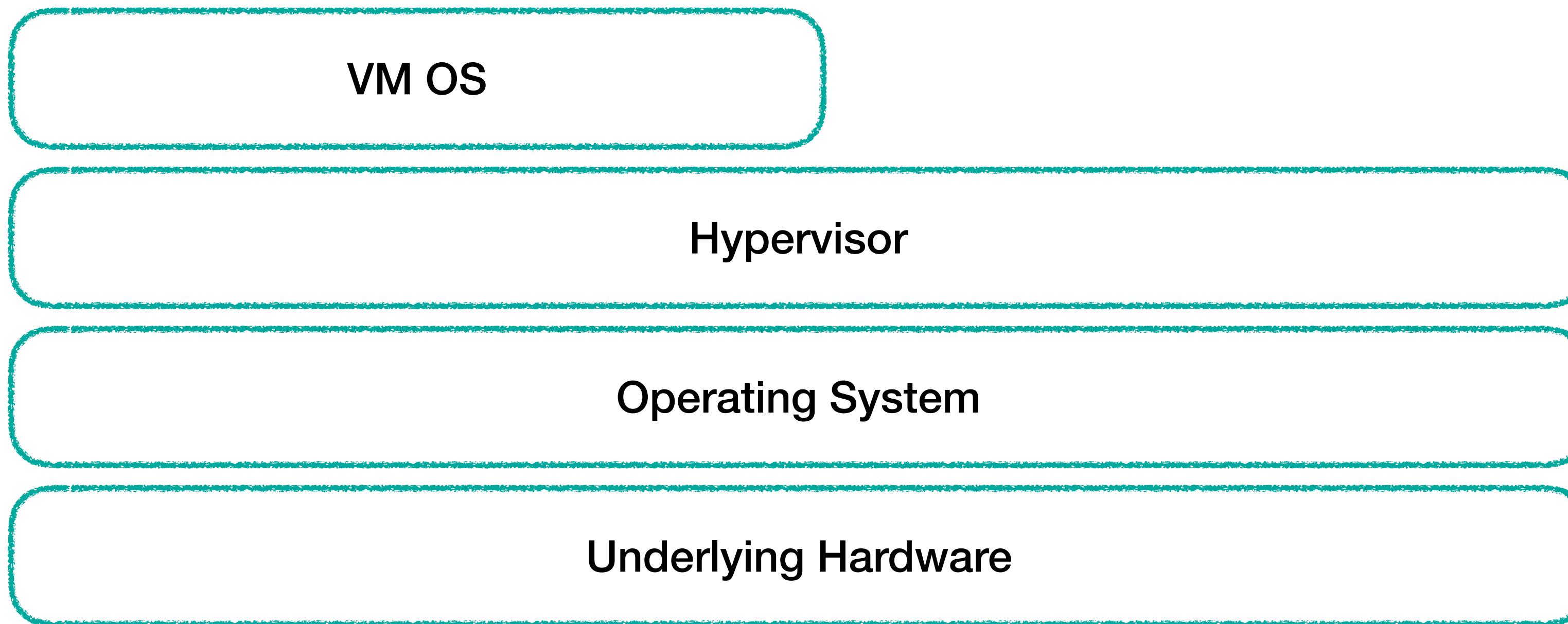
Operating System

Underlying Hardware

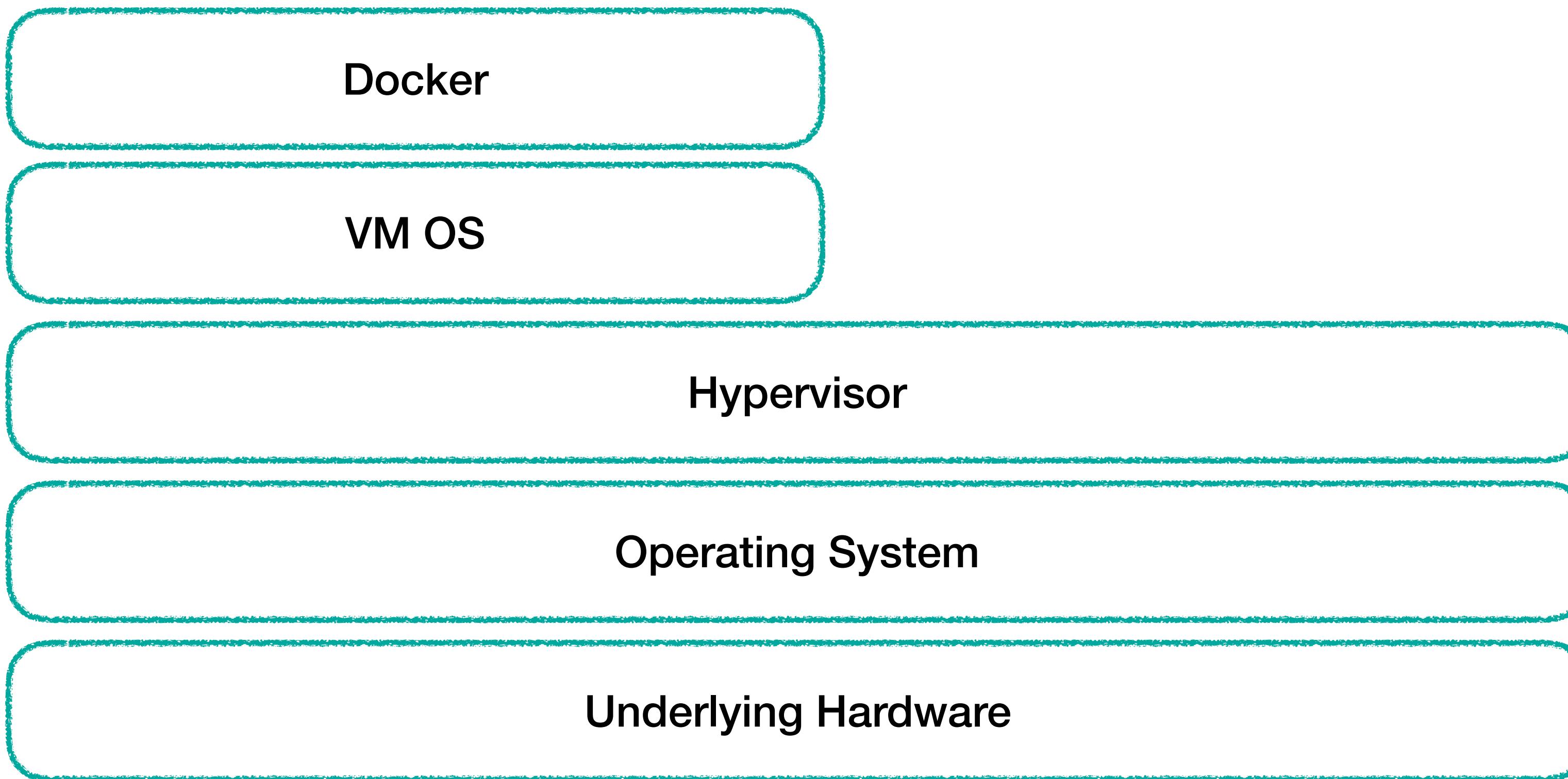
# PATCHING MADNESS!



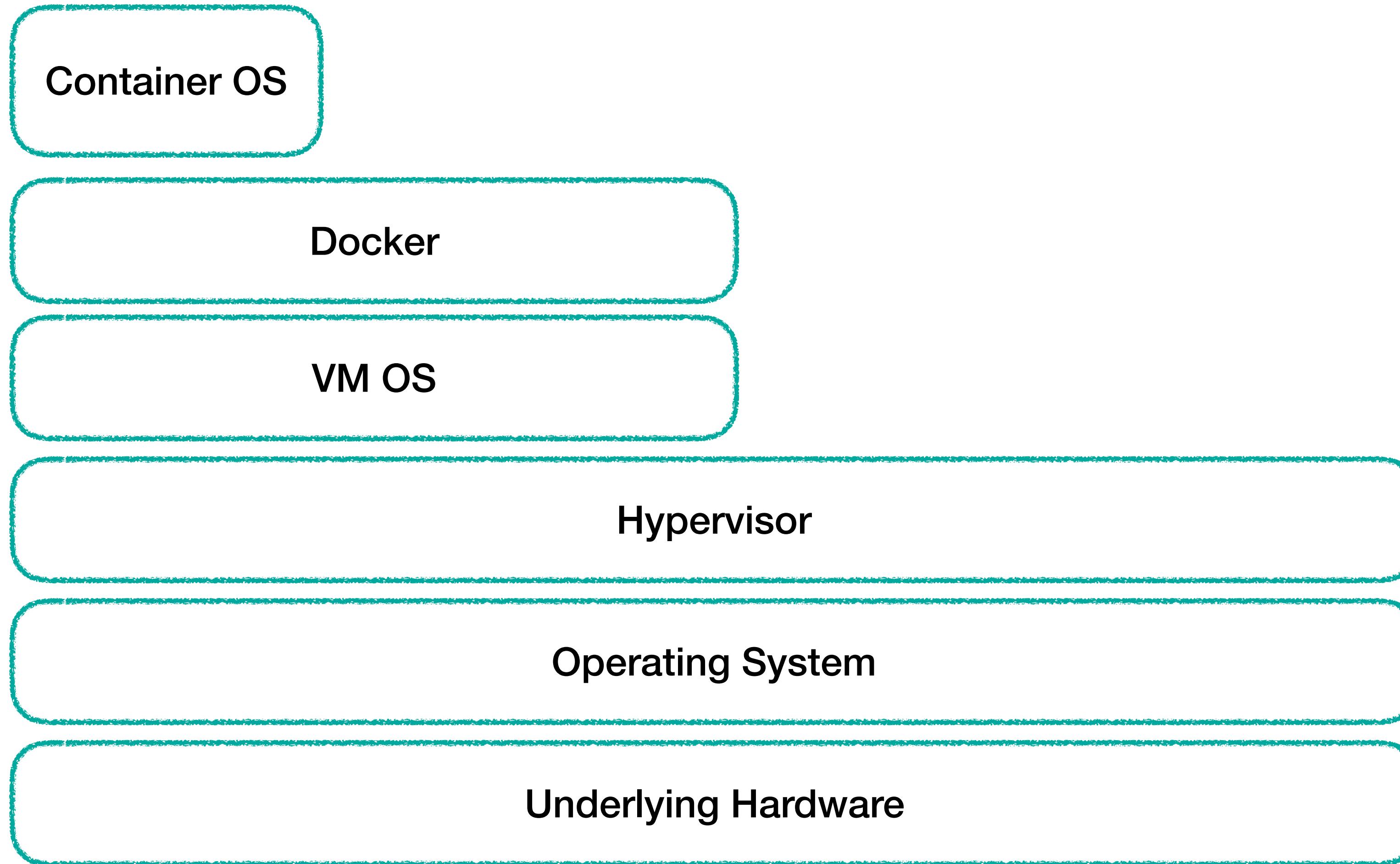
# PATCHING MADNESS!



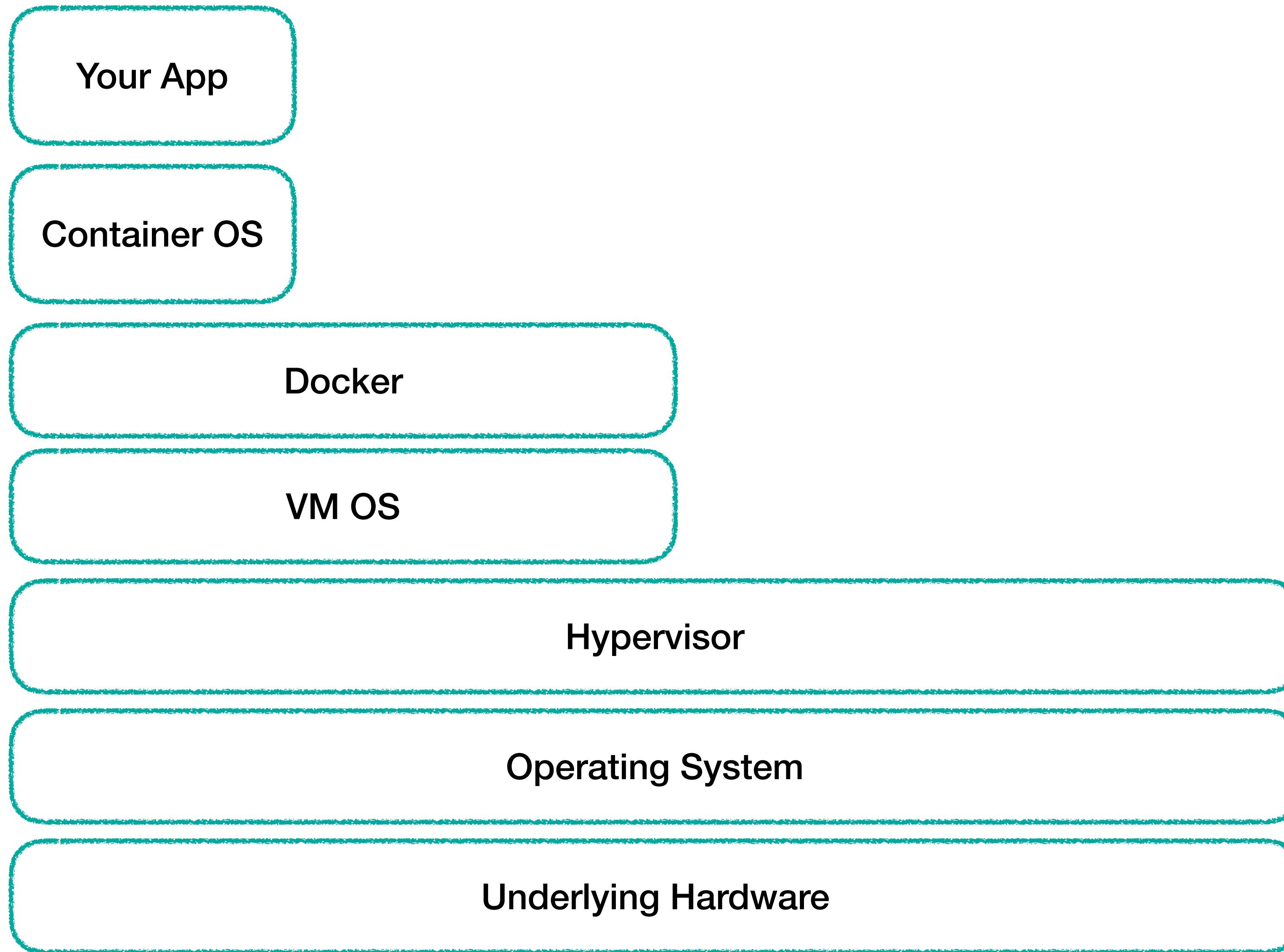
# PATCHING MADNESS!



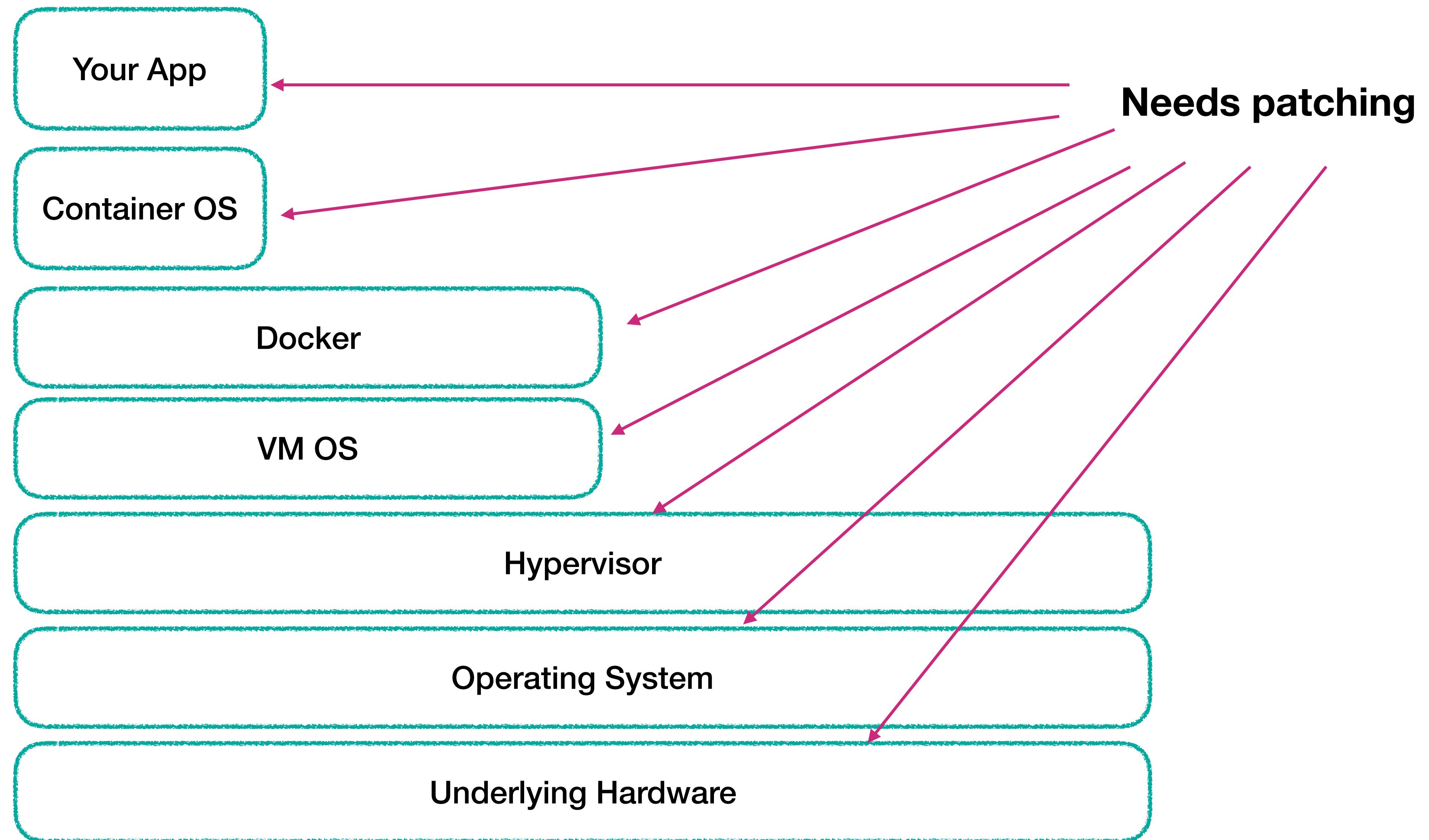
# PATCHING MADNESS!



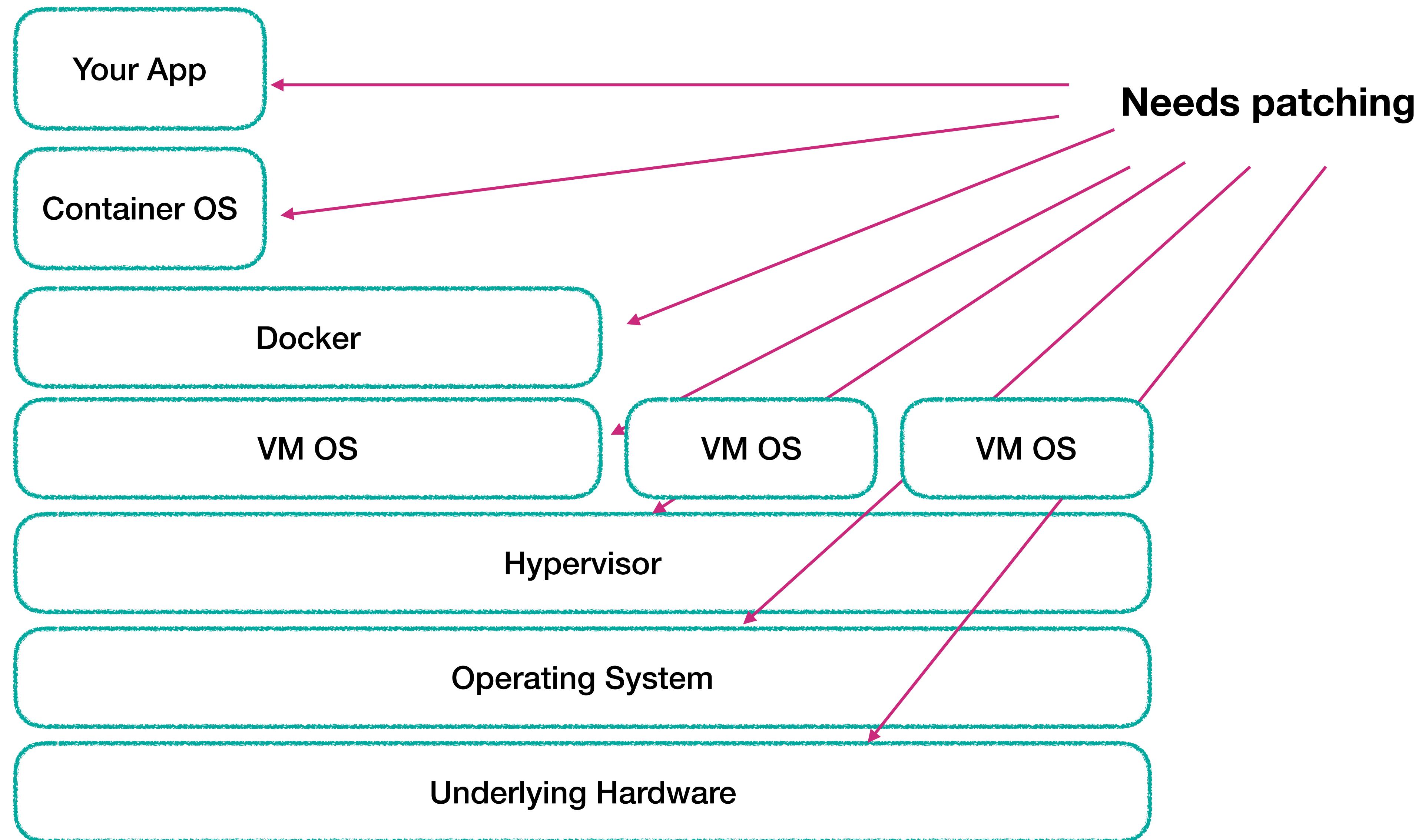
# PATCHING MADNESS!



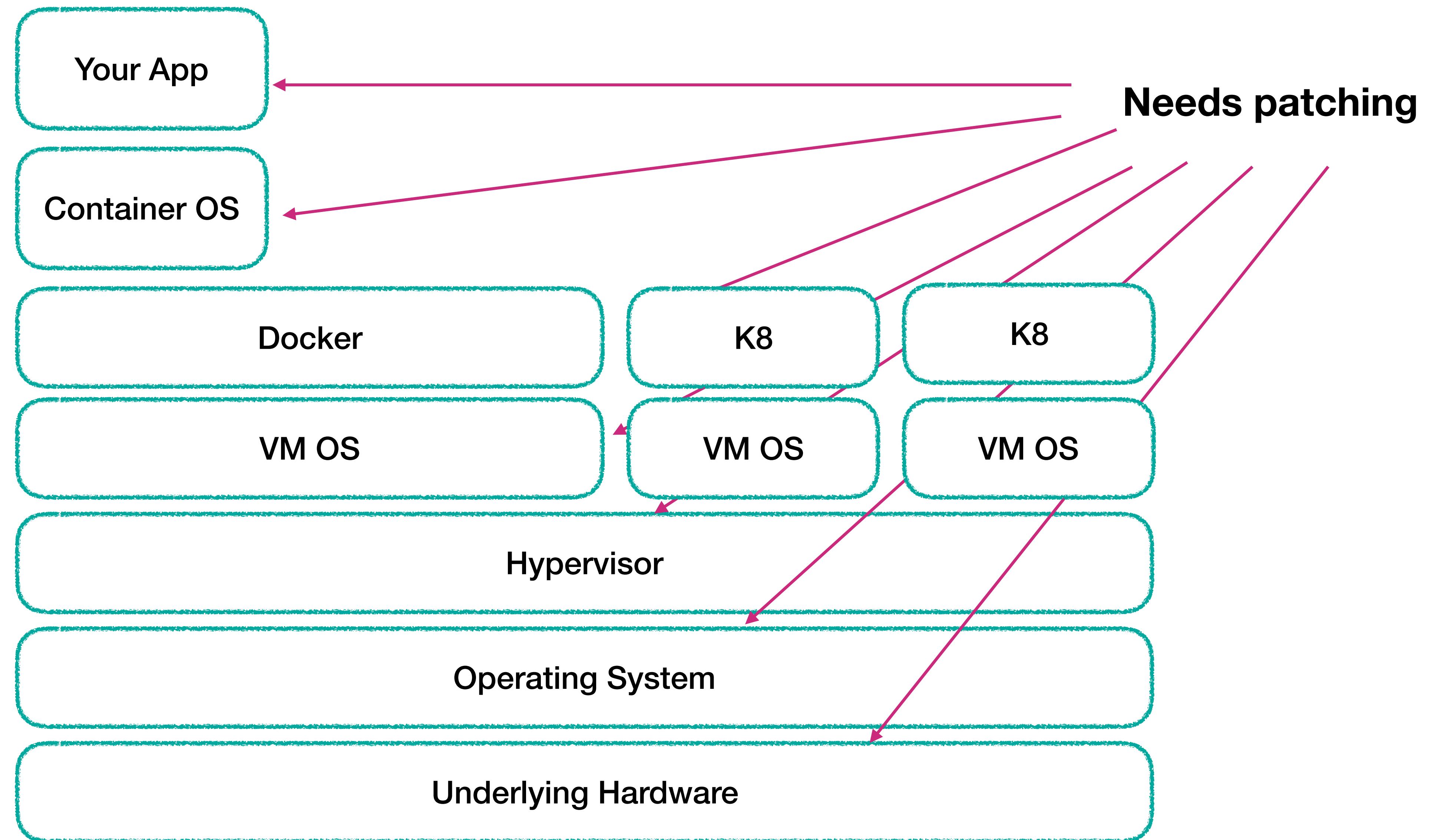
# PATCHING MADNESS!



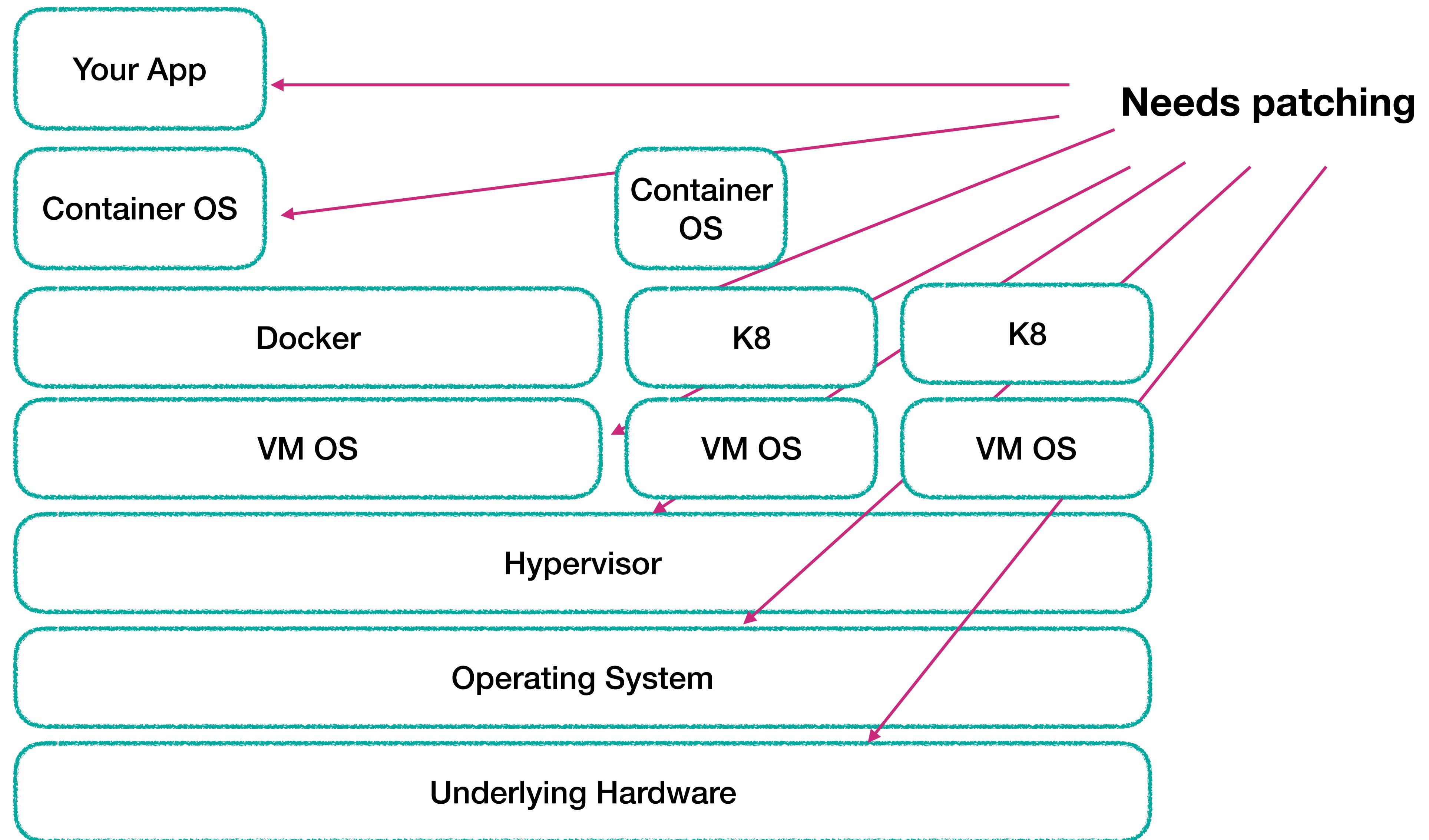
# PATCHING MADNESS!



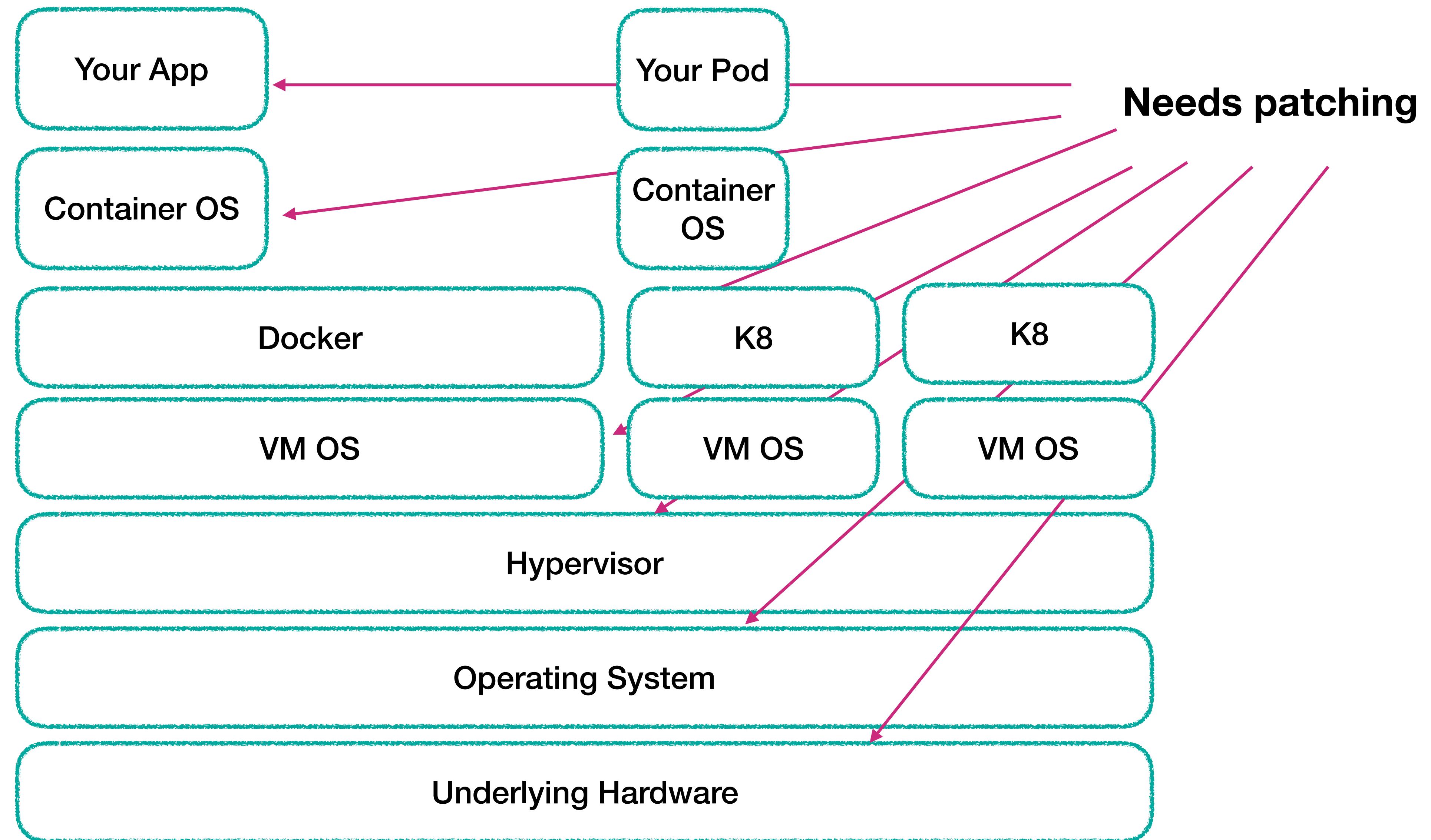
# PATCHING MADNESS!



# PATCHING MADNESS!



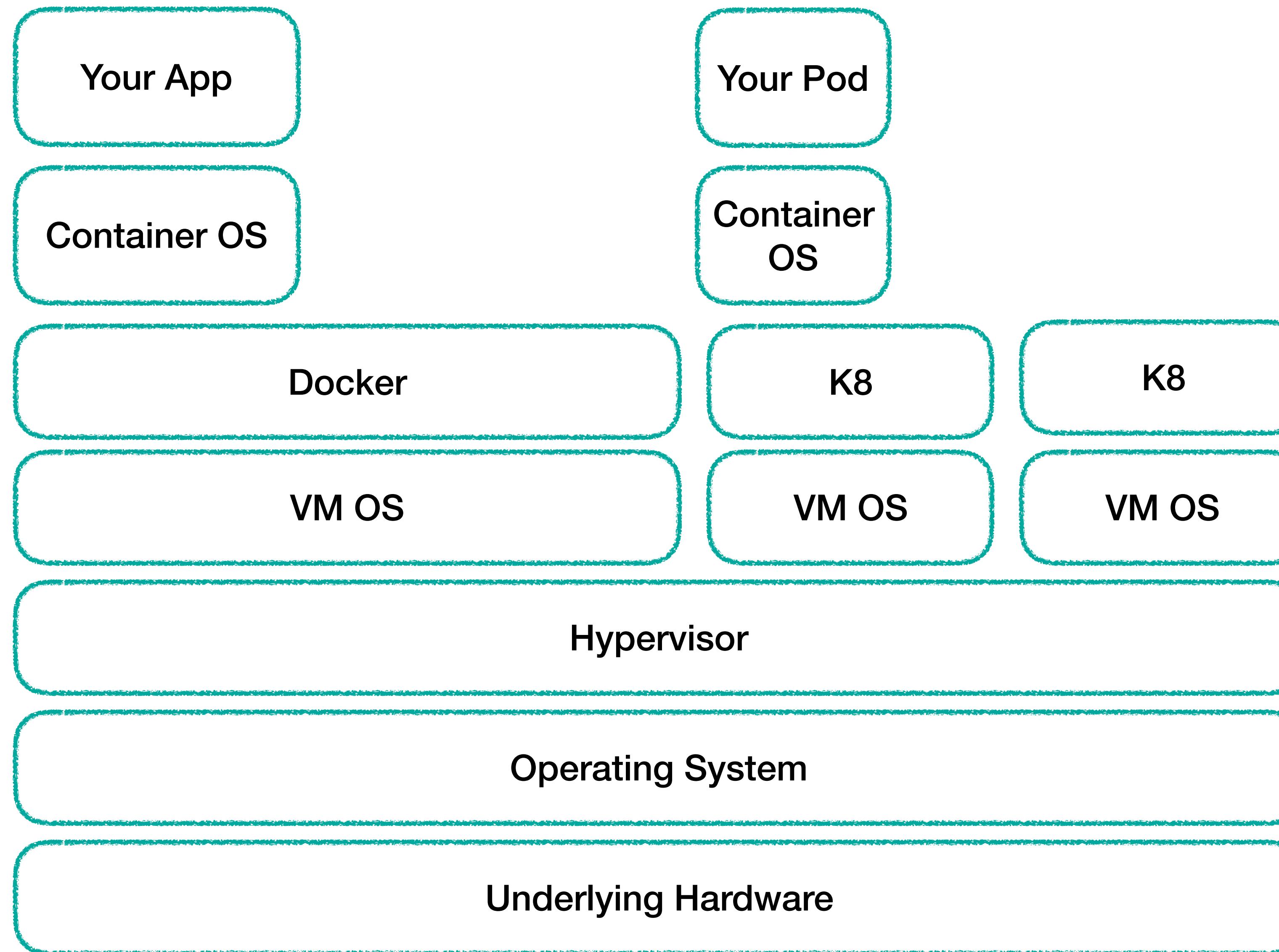
# PATCHING MADNESS!



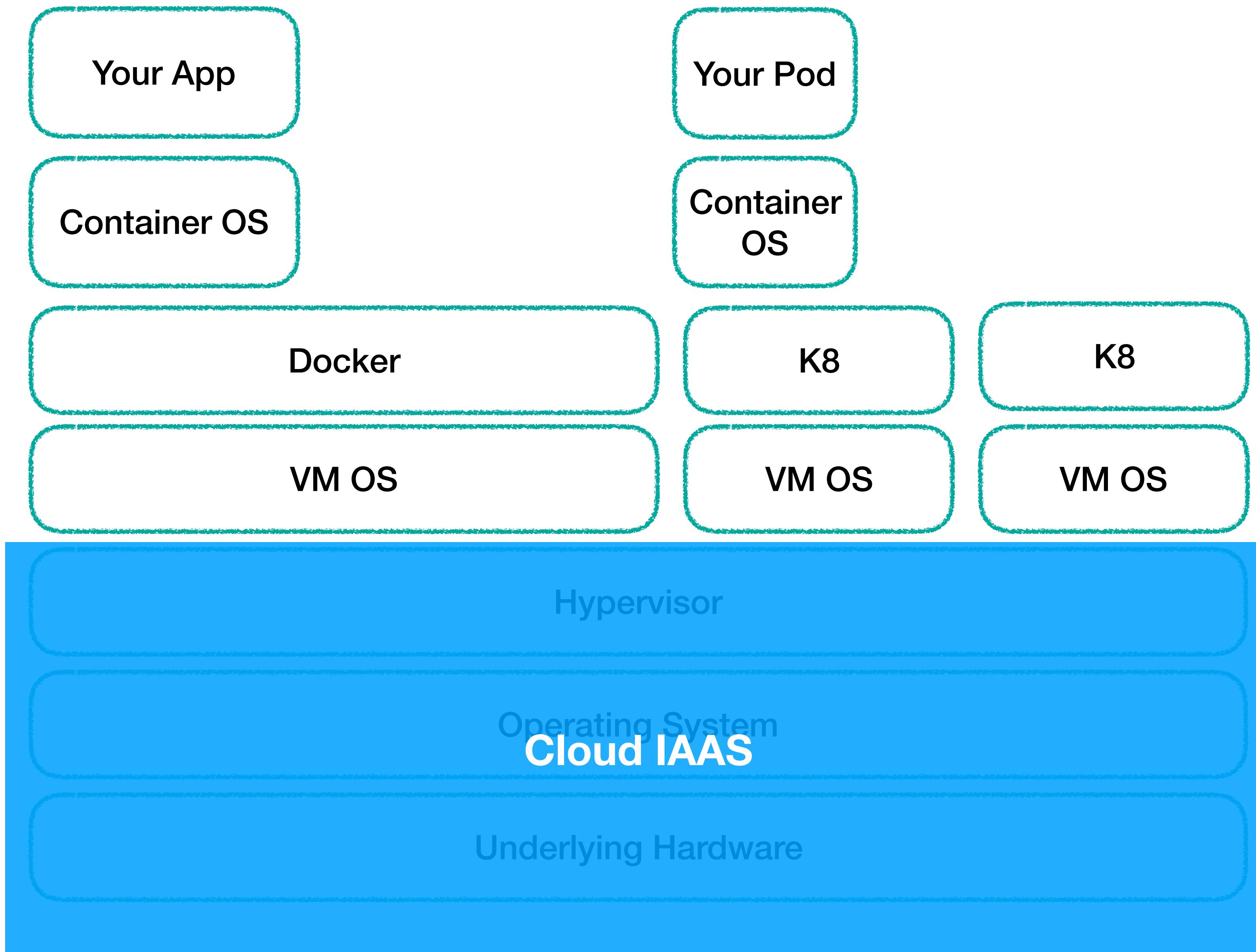
**So, how many of you are still sure you apply every patch within 2-4 months of them being found?**

**So what can you do about this?**

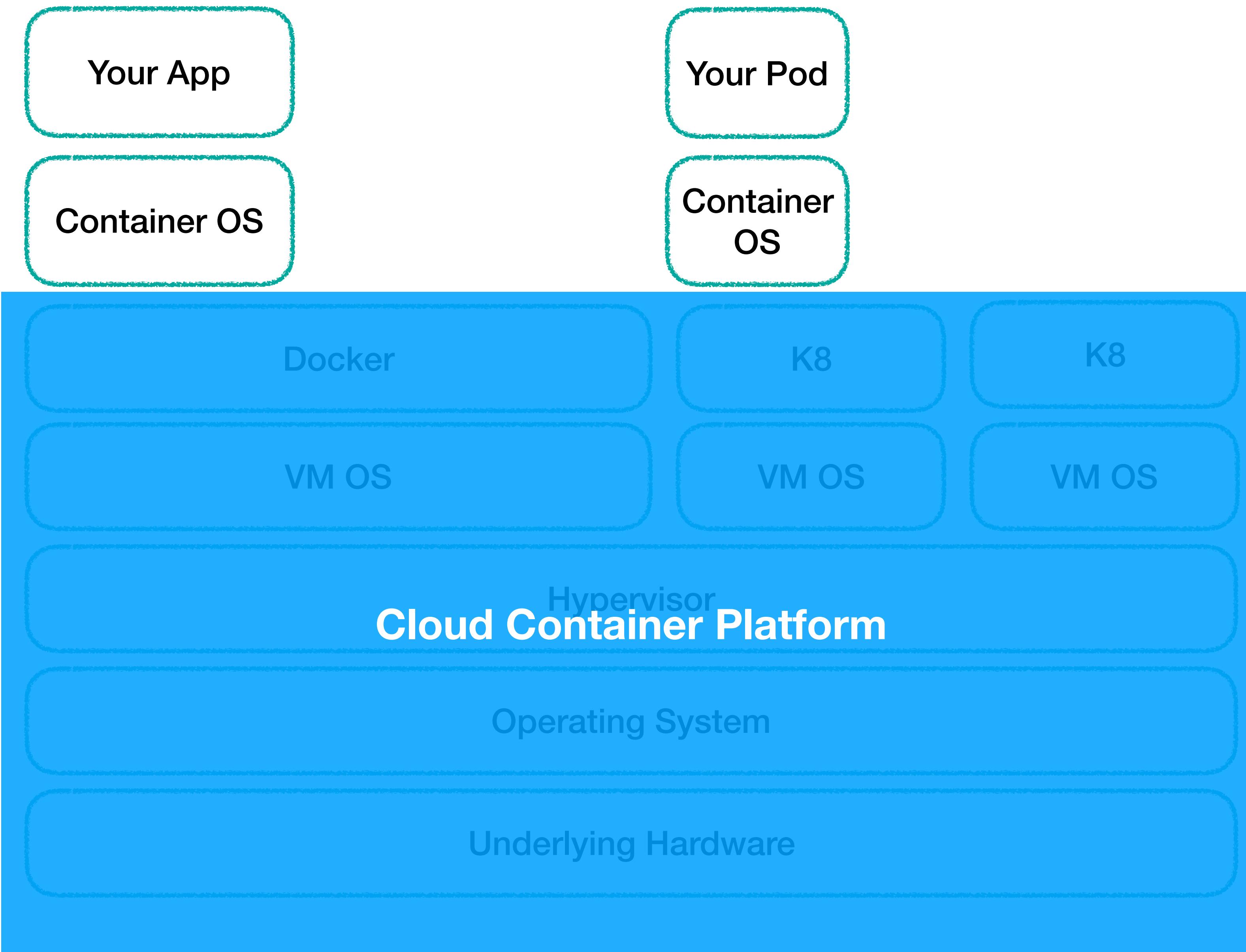
# BETTER ON THE CLOUD?



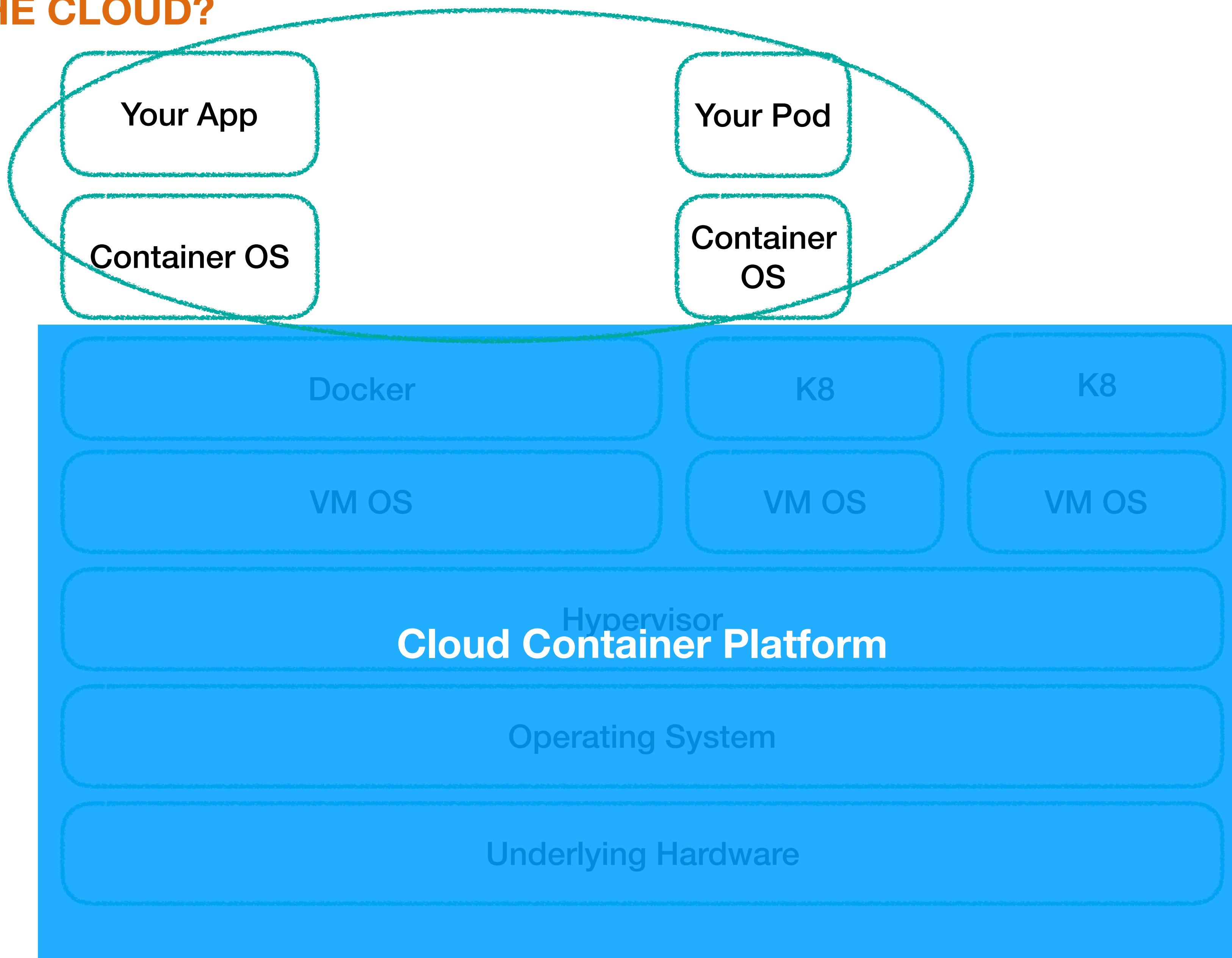
# BETTER ON THE CLOUD?



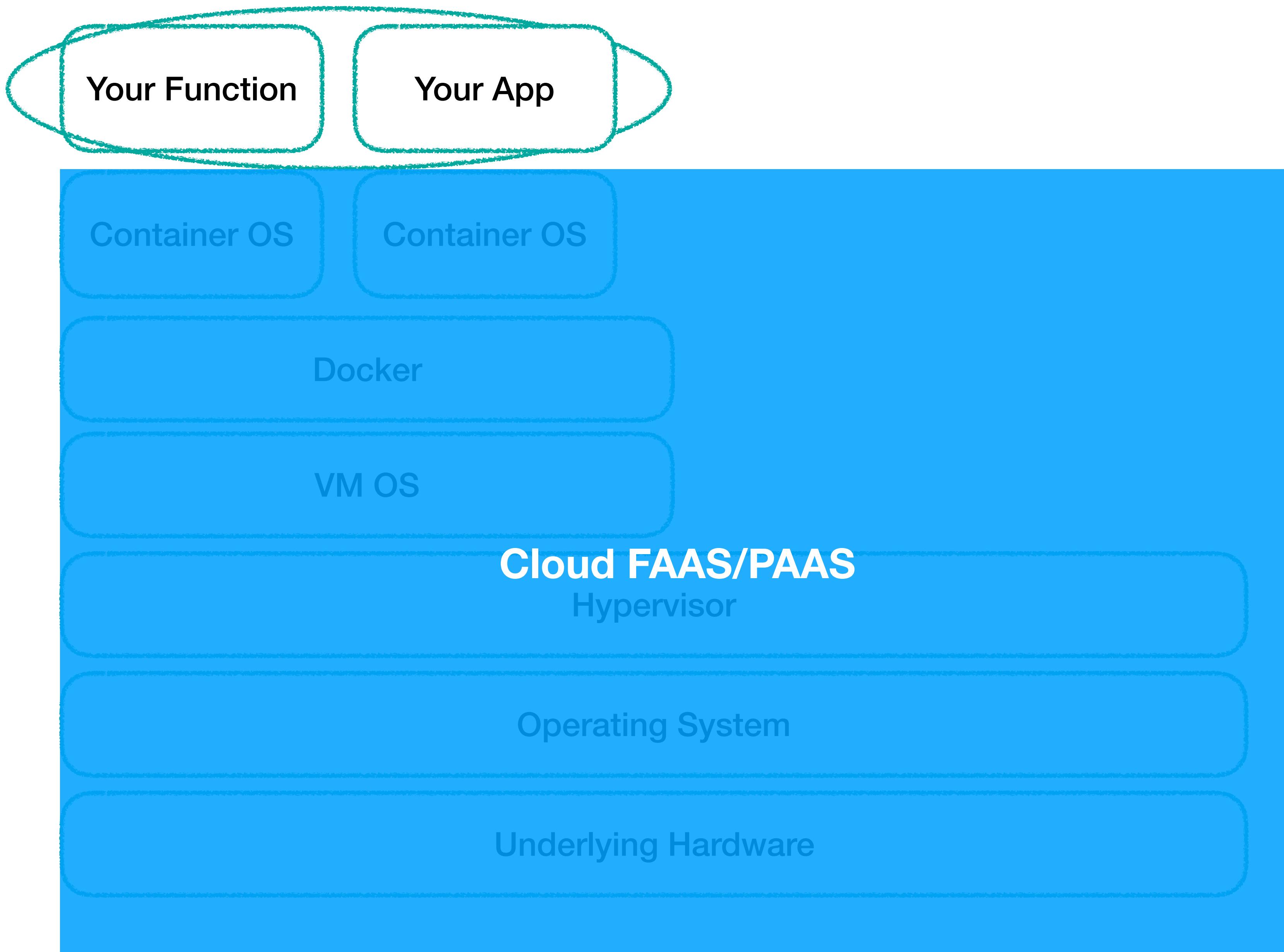
# BETTER ON THE CLOUD?



## BETTER ON THE CLOUD?



## BETTER WITH FAAS?



# CONTAINER SCANNING

README.md

## Clair

[build](#) [passing](#) [container](#) [ready](#) [go report](#) [A+](#) [godoc](#) [reference](#) [freenode](#) [#clair](#)

Note: The `master` branch may be in an *unstable* or even *broken* state during development. Please use [releases](#) instead of the `master` branch in order to get stable binaries.



# clair

Clair is an open source project for the [static analysis](#) of vulnerabilities in application containers (currently including [appc](#) and [docker](#)).

1. In regular intervals, Clair ingests vulnerability metadata from a configured set of sources and stores it in the database.
2. Clients use the Clair API to index their container images; this creates a list of *features* present in the image and stores them in the database.
3. Clients use the Clair API to query the database for vulnerabilities of a particular image; correlating vulnerabilities and features is done for each request, avoiding the need to rescan images.
4. When updates to vulnerability metadata occur, a notification can be sent to alert systems that a change has occurred.

Our goal is to enable a more transparent view of the security of container-based infrastructure. Thus, the project was named `Clair` after the French term which translates to *clear, bright, transparent*.

<https://github.com/coreos/clair>

@samnewman

## CONTAINER SCANNING (CONT)

The screenshot shows the aqua container scanning interface. The left sidebar includes links for Dashboard, Images, Containers, Applications, and Audit under the heading "Administration". The main content area displays the "Repositories & Images" section for "wordpress:latest". It features an "Image Overview" with a legend: High (red), Medium (orange), Low (yellow), and Average Score (green). Below this, two container images are compared: "wordpress:latest" and "php:5.6-apache". Each image has its own set of vulnerability counts and a detailed breakdown. A search bar is present at the bottom right. A table below lists specific vulnerabilities with columns for Name, Score, Severity, and Publish Date.

Name	Score	Severity	Publish Date
> CVE-2013-4482	10	High	2016
> CVE-2016-1751	10	High	2016
> CVE-2014-5485	10	High	2015-01-19

<https://www.aquasec.com>

# MONITOR OUTDATED DEPENDENCIES

The screenshot shows the Snyk website homepage. At the top, there's a dark header with the Snyk logo, navigation links (Test, Vulnerability DB, Docs, Blog, Features, Partners, Pricing, Login), social media icons (GitHub, Bitbucket, Google), and a "Signup" button. Below the header is a large banner featuring a stylized dog head logo and the text "Snyk continuously finds and fixes vulnerabilities in your dependencies." followed by "Protect and monitor your JavaScript, Ruby and Java apps".

**Source code protection**  
Find vulnerabilities in your GitHub repositories and remediate risks with updates and patches. Add Snyk to your CI/CD process with support for Jenkins, Circle CI, Travis and more.

**New! Serverless & PaaS monitoring**  
Continuously monitor your function apps. Get toxic security alerts and deploy critical updates. Support for Heroku, AWS Lambda and more.

[Quick start with GitHub](#) [Sign up for a free account](#)

**Integrations:** GitHub, Bitbucket, Travis CI, Jenkins, Heroku, AWS Lambda

**Statistics:** 83% of Snyk users found security issues in their dependencies. We can help you find, fix, and prevent vulnerabilities.

- Automatically test your application dependencies
- Fix security risks with upgrades and patches
- Prevent you from adding vulnerable dependencies
- Stay alert about new vulnerabilities

[Find out how](#)

A small inset image in the bottom right corner shows a screenshot of the Snyk dashboard interface, displaying a list of dependencies with their status (e.g., Up-to-date, Vulnerable) and severity levels (red, yellow, green).

<https://snyk.io/>

@samnewman

## AUTOMATICALLY PATCH APP DEPENDENCIES

[Snyk Update] New fixes for 25 vulnerable dependency paths #1

[Open](#) snyk-bot wants to merge 1 commit into master from snyk-fix-836b436e

Conversation 0 Commits 1 Files changed 2

 snyk-bot commented an hour ago First-time contributor +  

This project has vulnerabilities that could not be fixed, or were patched when no upgrade was available. Good news, new upgrades or patches have now been published! This pull request fixes vulnerable dependencies you couldn't previously address.

The PR includes:

- Changes to package.json to upgrade the vulnerable dependencies to a fixed version.

<https://snyk.io/>

@samnewman

# DO SOME THREAT MODELLING

The screenshot shows the Microsoft Security Development Lifecycle (SDL) website. At the top, there's a banner with three men and the title "Life in the Digital Crosshairs". Below the banner, there's a section titled "What is the Security Development Lifecycle ?" featuring a green hexagonal icon with a white circle and text about the SDL process. A horizontal timeline shows phases: Training, Requirements, Design (highlighted in green), Implementation, Verification, Release, and Response. A link "Click to select a phase" is below the timeline. The "Design Phase" is currently selected, with a green header and the sub-section "SDL Practice #5: Establish Design Requirements" and a note about minimizing risk. To the right, there's a sidebar titled "Assess your security" with a link to "Get Started" and a section titled "Tools" listing four security analysis tools: Attack Surface Analyzer 1.0, Microsoft Threat Modeling Tool 2014, MiniFuzz basic file fuzzing tool, and Regular expression file fuzzing tool.

Life in the Digital Crosshairs

Experience the Untold Story

Learn more →

What is the Security Development Lifecycle ?

The Security Development Lifecycle (SDL) is a software development process that helps developers build more secure software and address security compliance requirements while reducing development cost

Training Requirements Design Implementation Verification Release Response

Click to select a phase

Design Phase

SDL Practice #5: Establish Design Requirements

Considering security and privacy concerns early helps minimize the risk of schedule disruptions and reduce a project's expense.

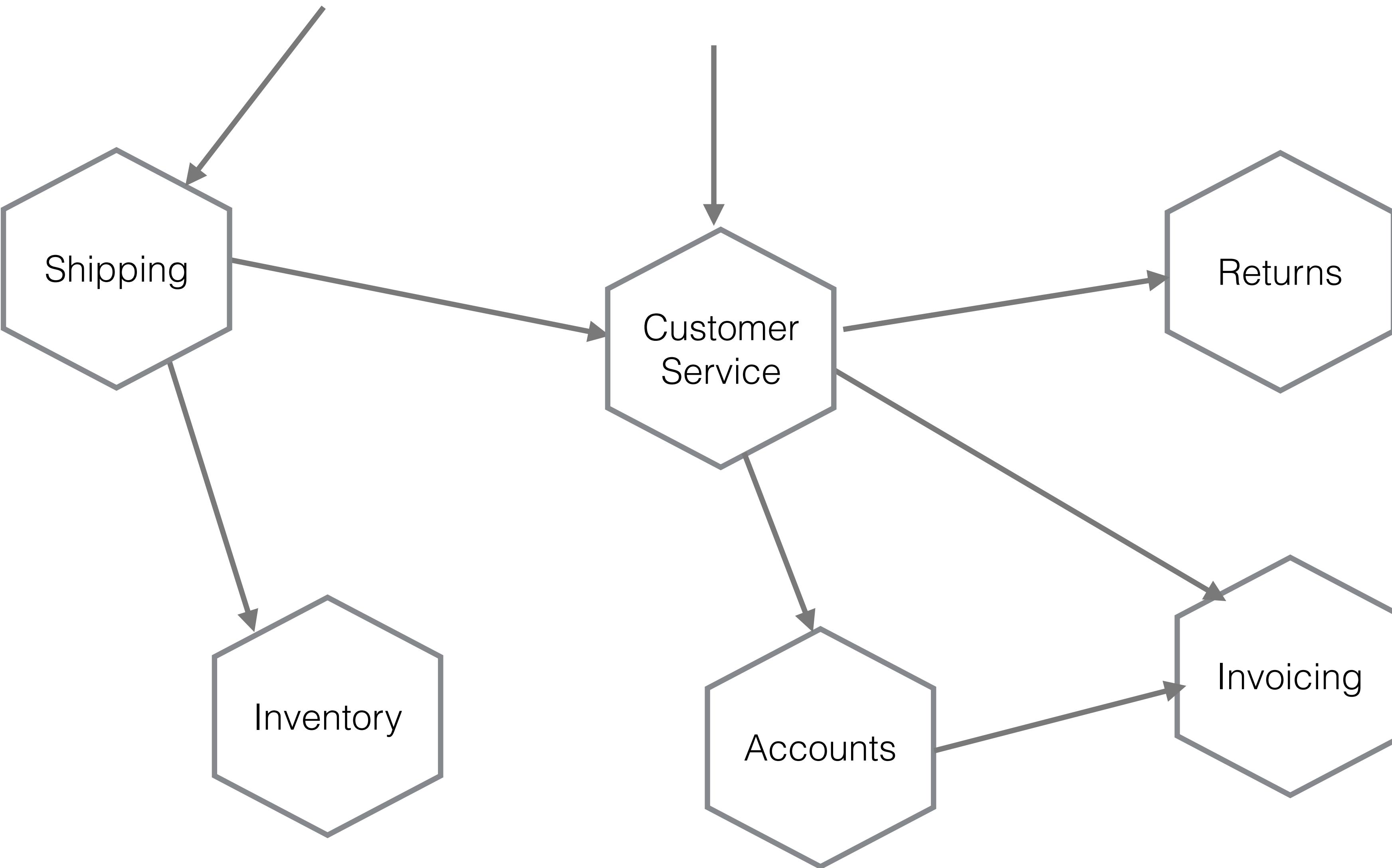
Assess your security

Get Started

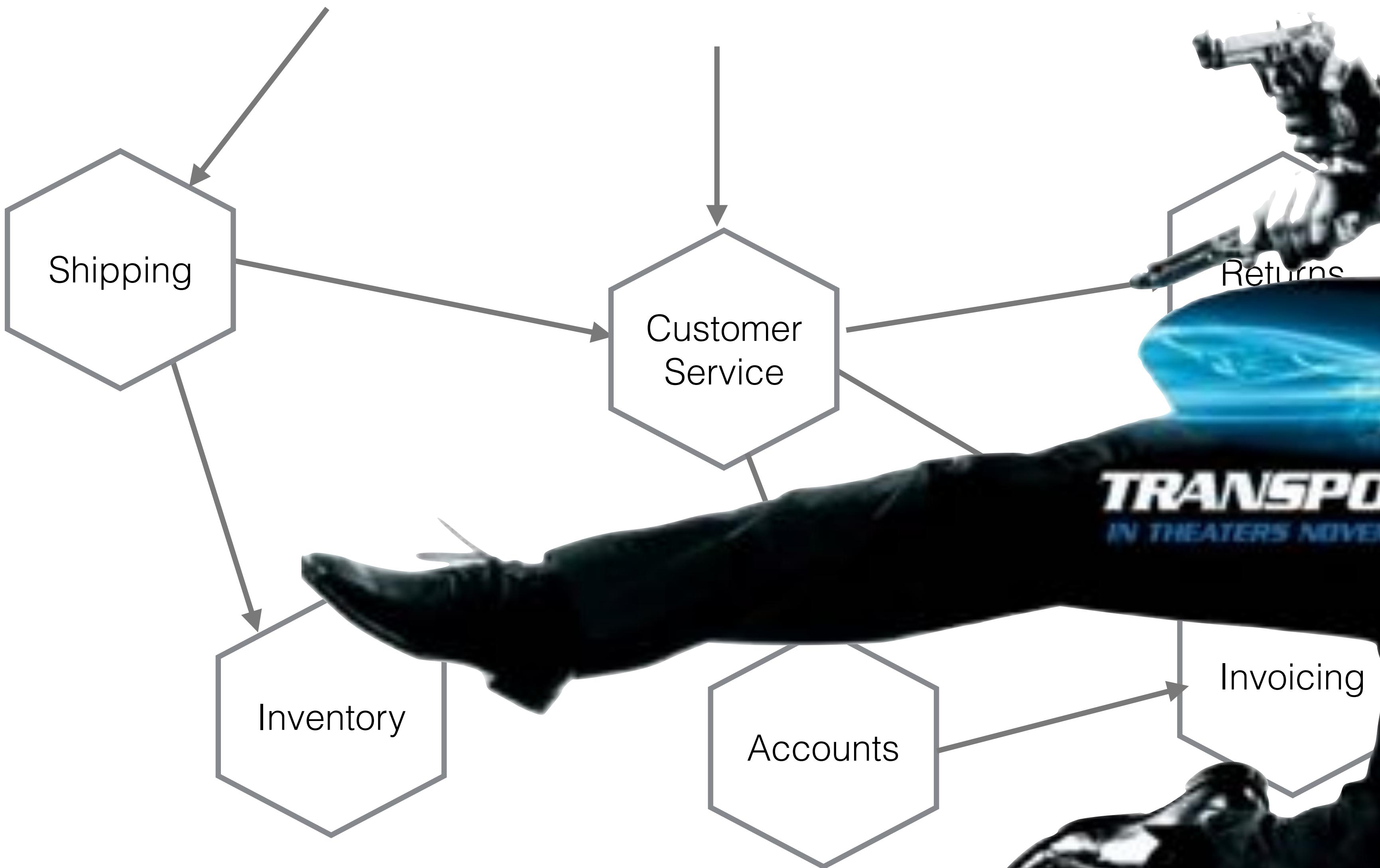
Tools

- Attack Surface Analyzer 1.0  
Understand your attack surface before & after new apps are deployed.
- Microsoft Threat Modeling Tool 2014  
A tool to help engineers find and address system security issues.
- MiniFuzz basic file fuzzing tool  
A simple fuzzer designed to ease adoption of fuzz testing.
- Regular expression file fuzzing tool  
A tool to test for potential denial of service vulnerabilities.

<https://www.microsoft.com/en-us/sdl/>



J A S O N S T A T H A M

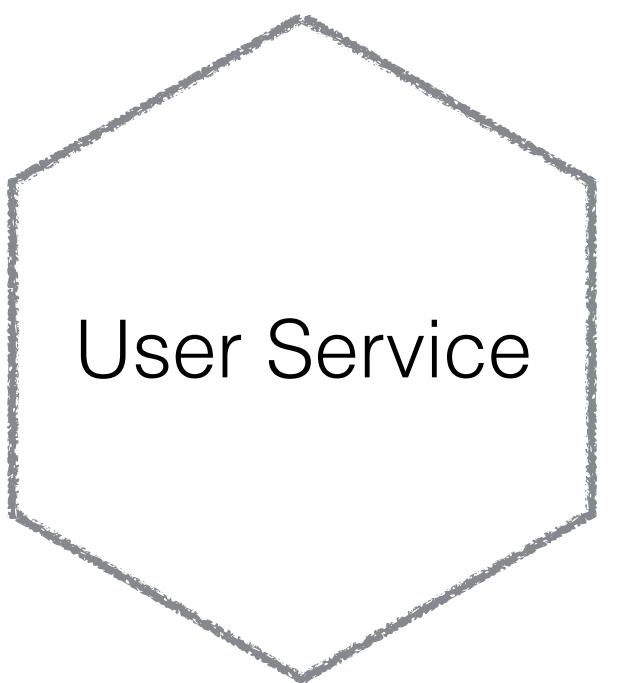


# MUSIC CORP 2018

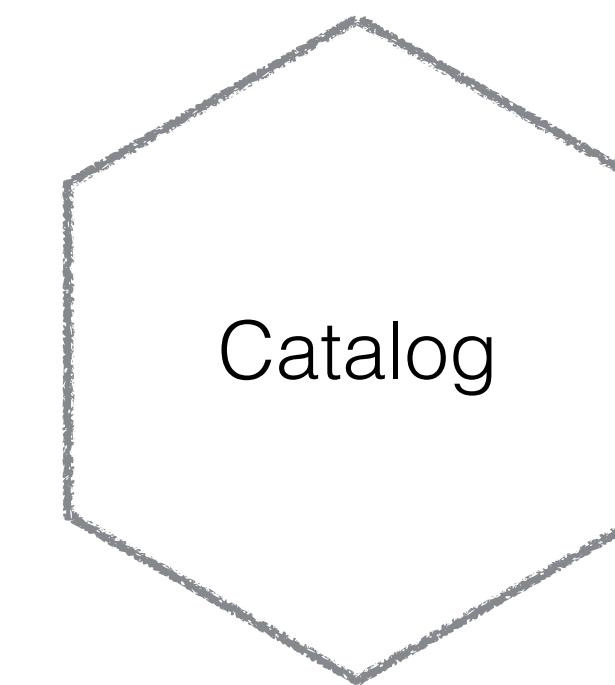
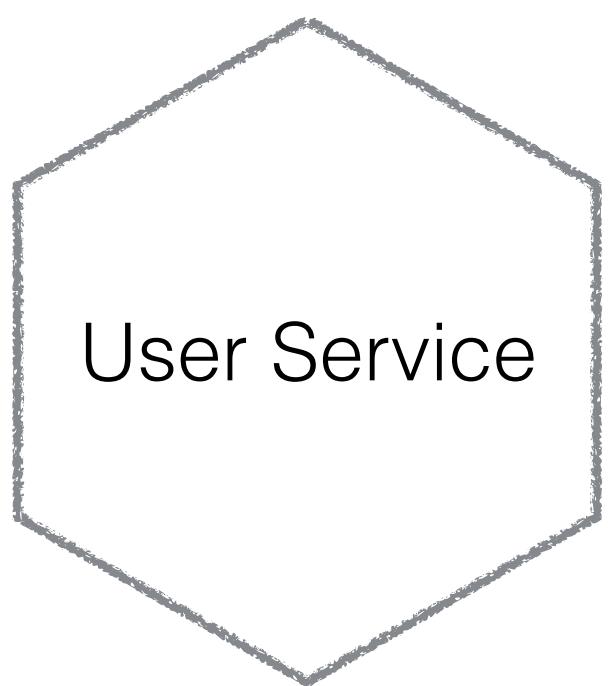
# MUSIC CORP 2018



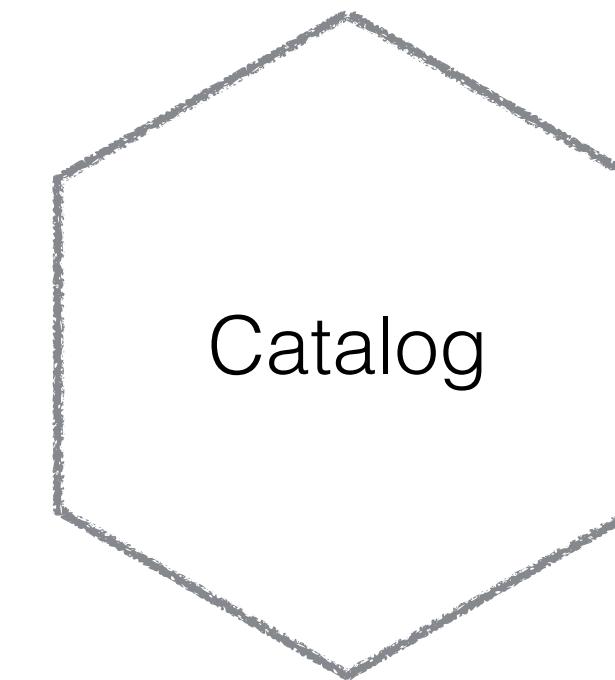
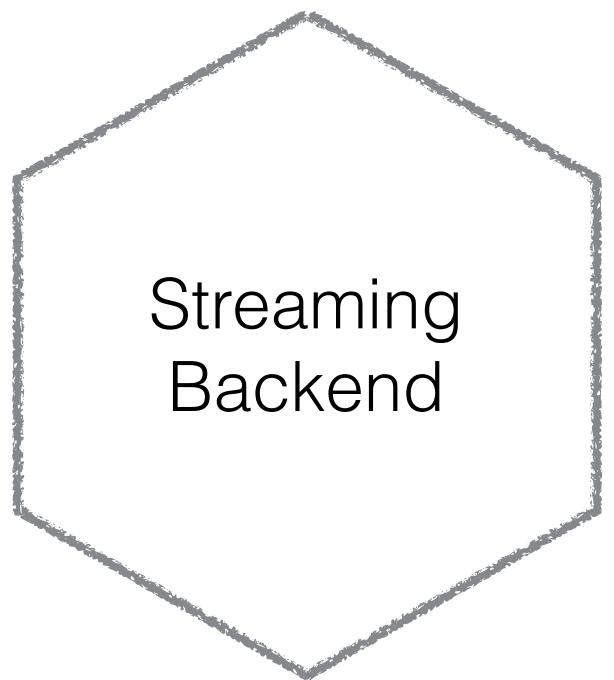
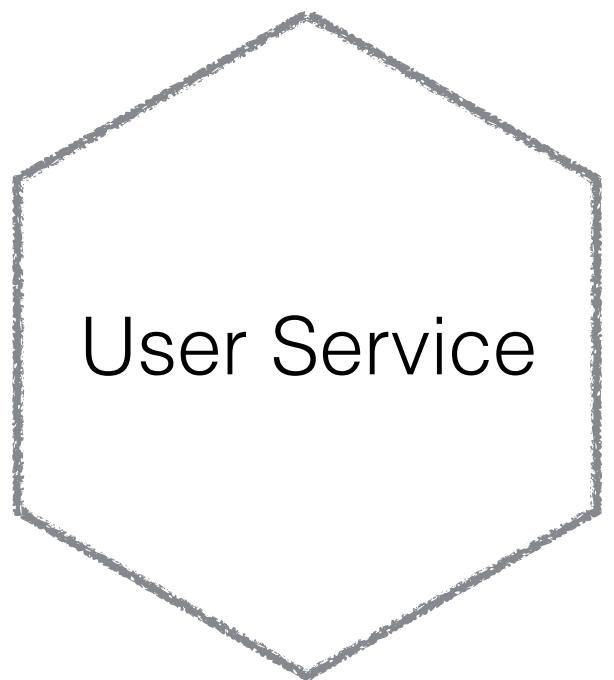
# MUSIC CORP 2018



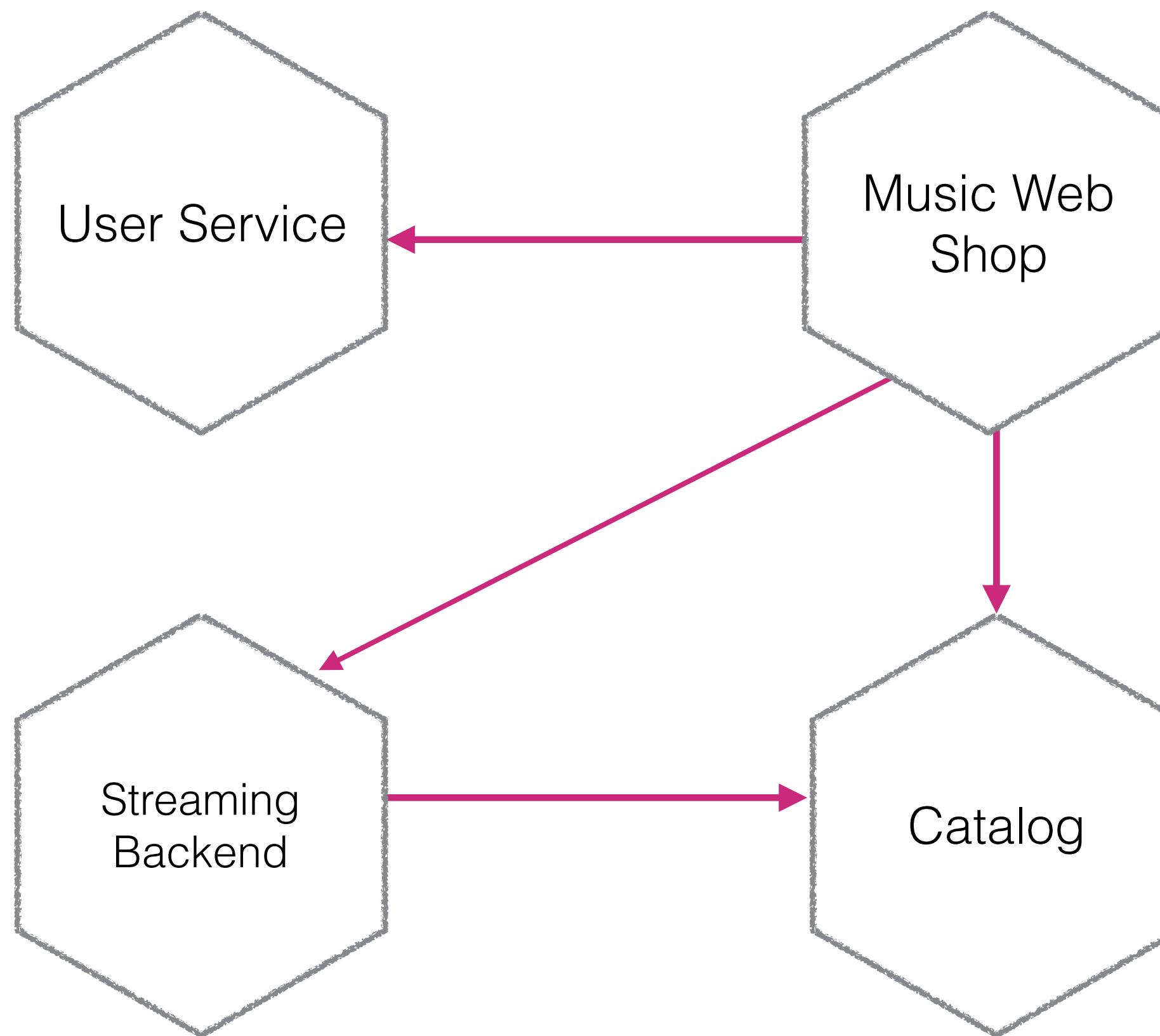
# MUSIC CORP 2018



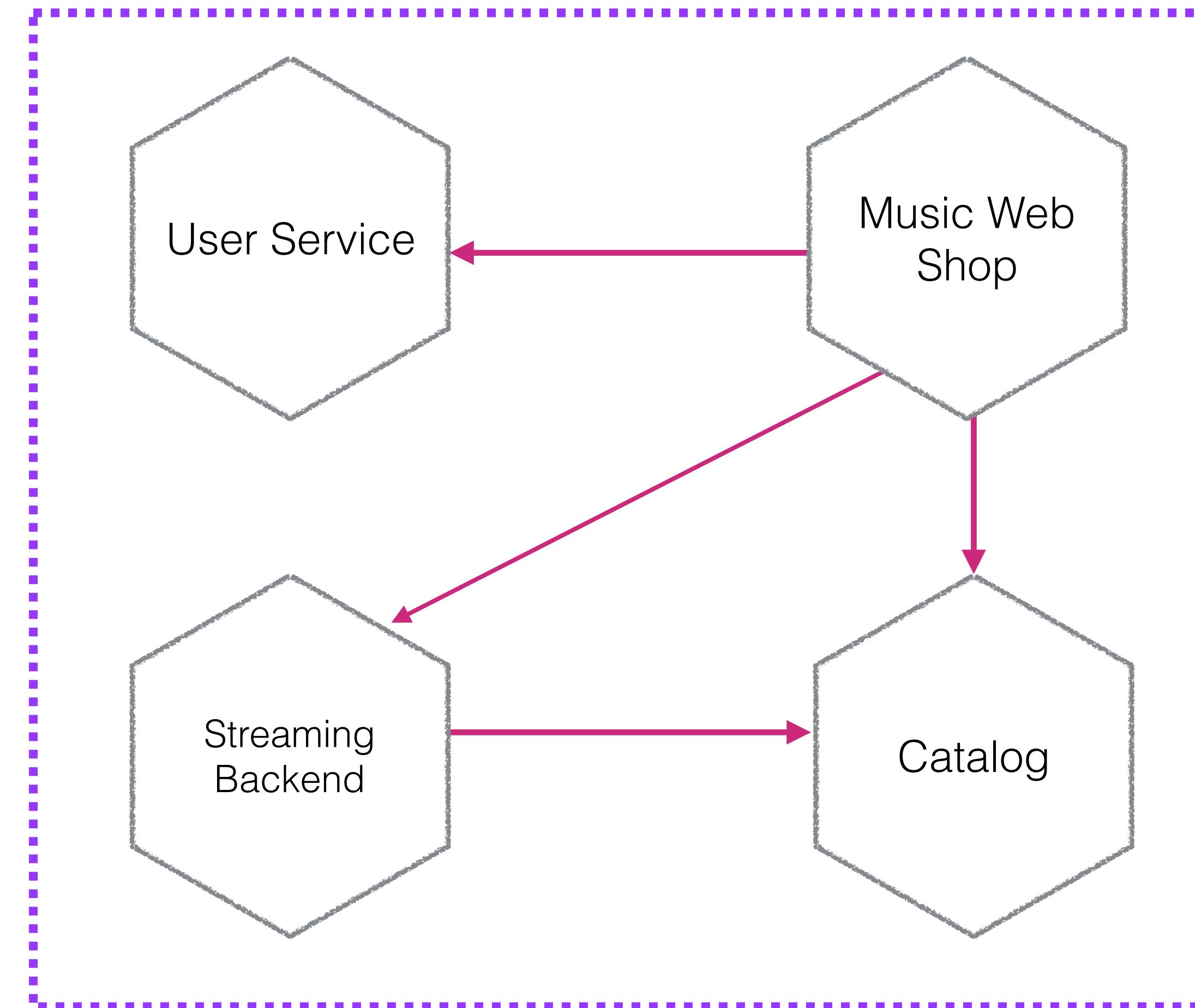
# MUSIC CORP 2018



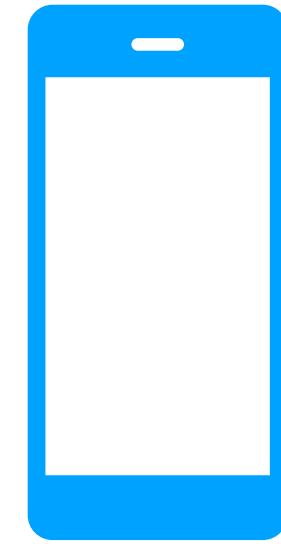
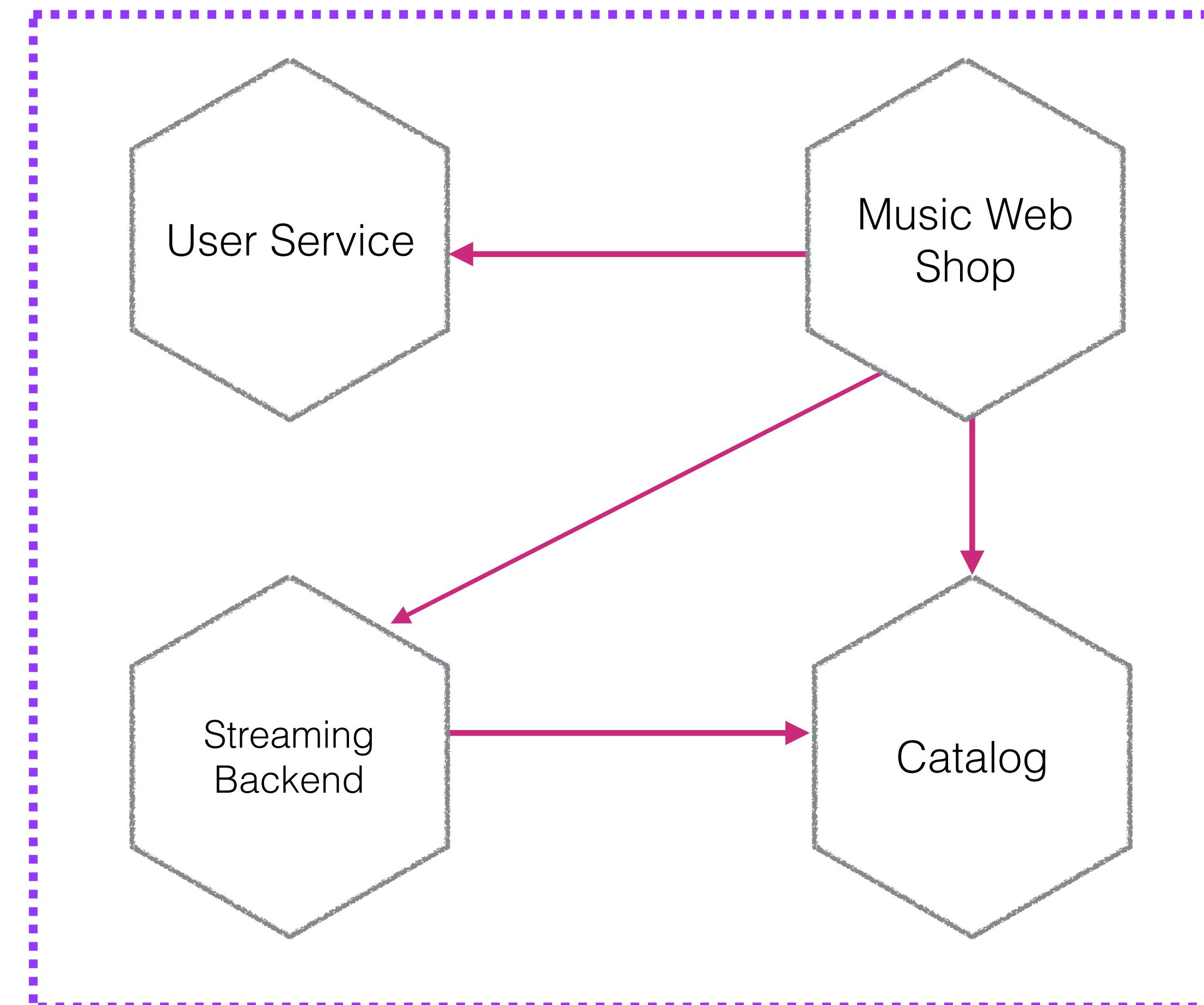
# MUSIC CORP 2018



# MUSIC CORP 2018

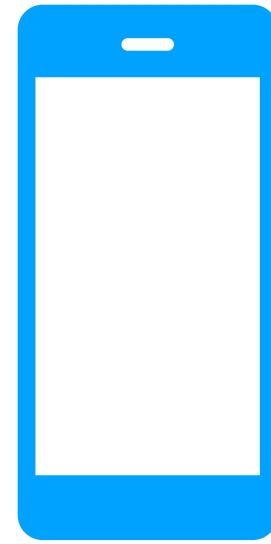
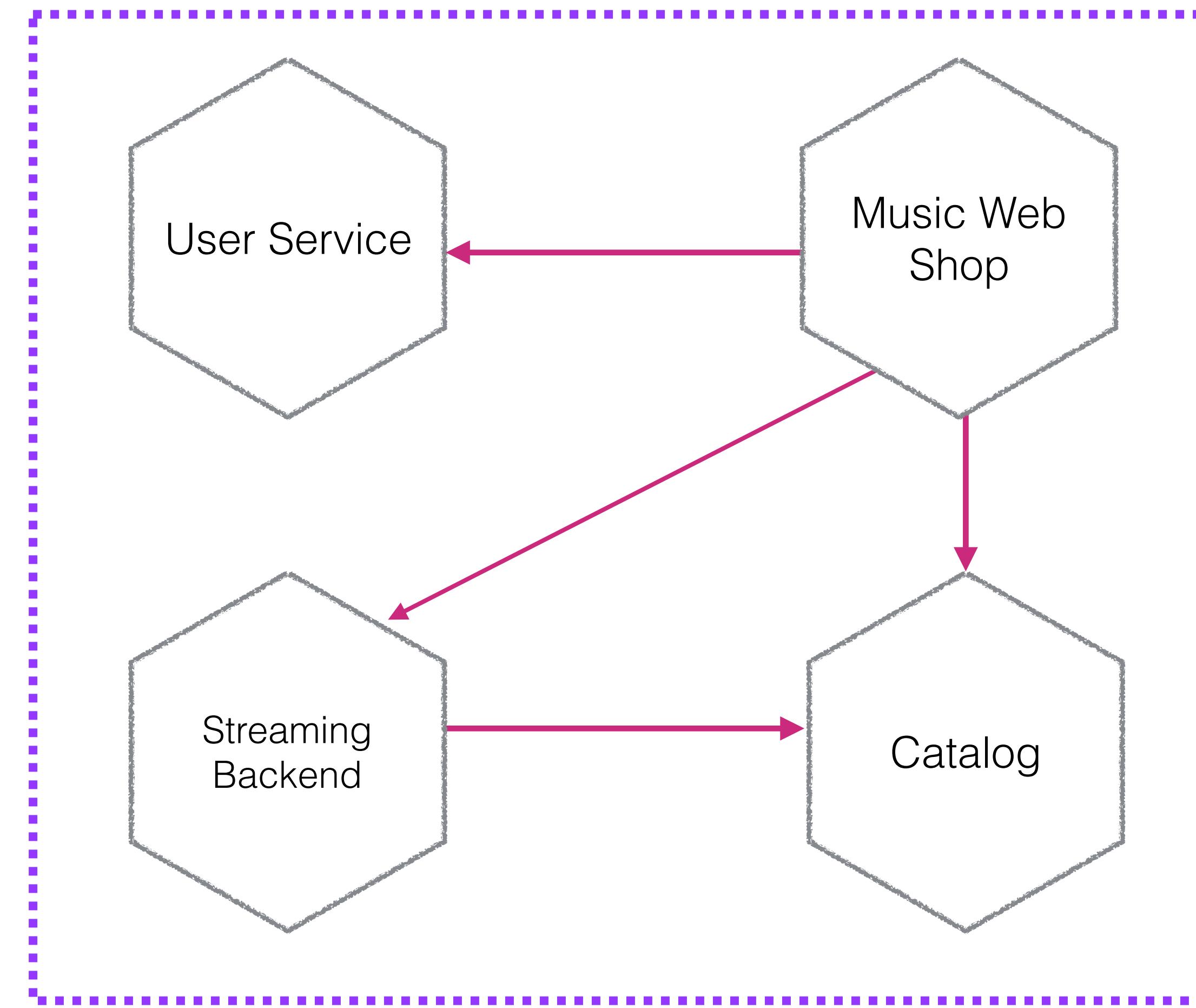


# MUSIC CORP 2018

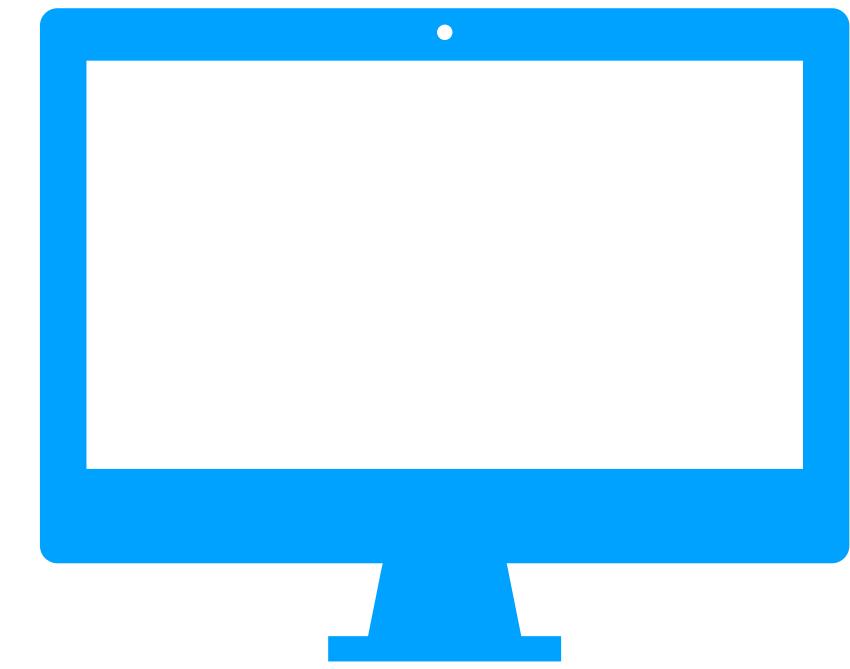


Native Mobile

# MUSIC CORP 2018

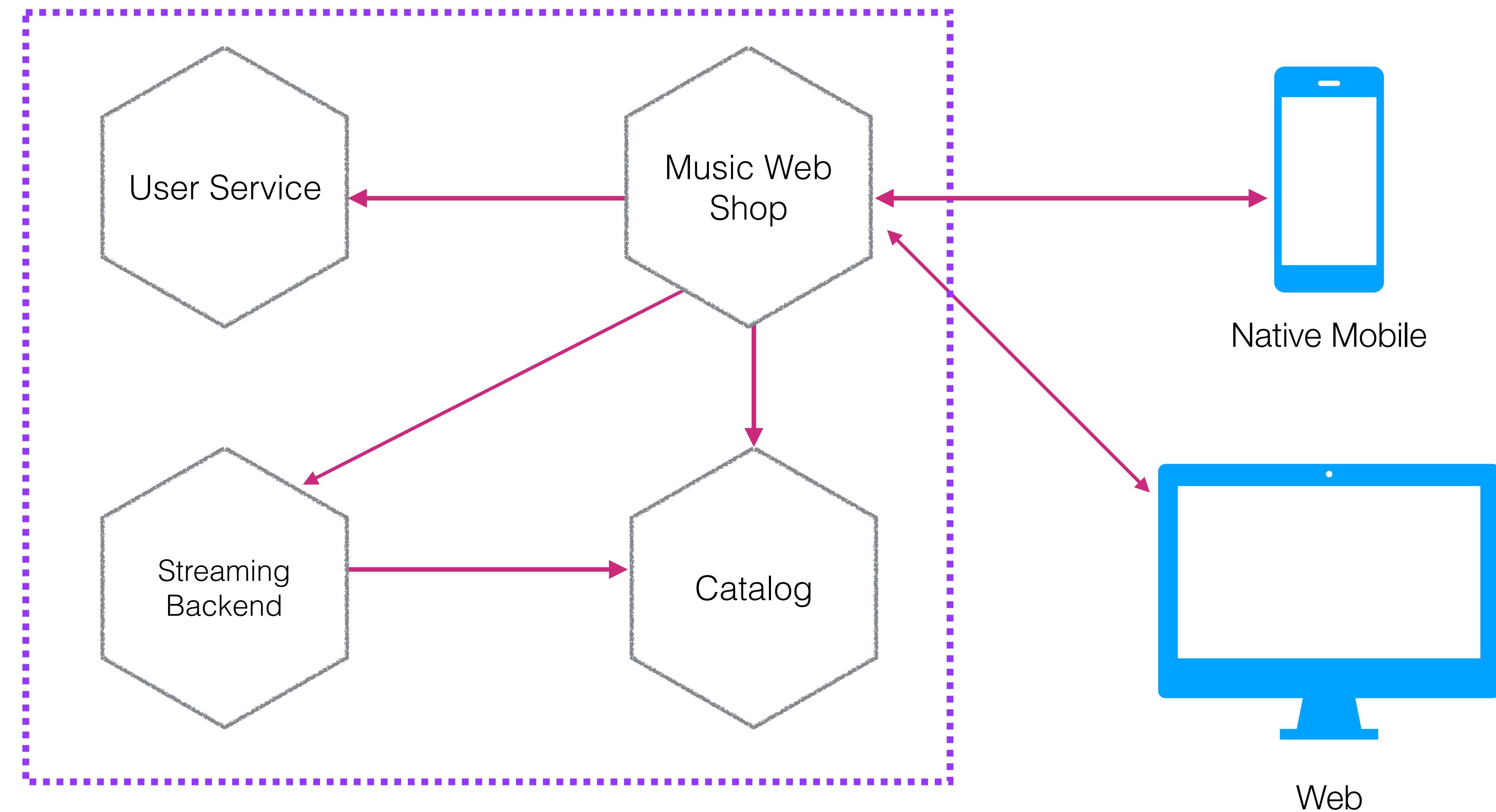


Native Mobile

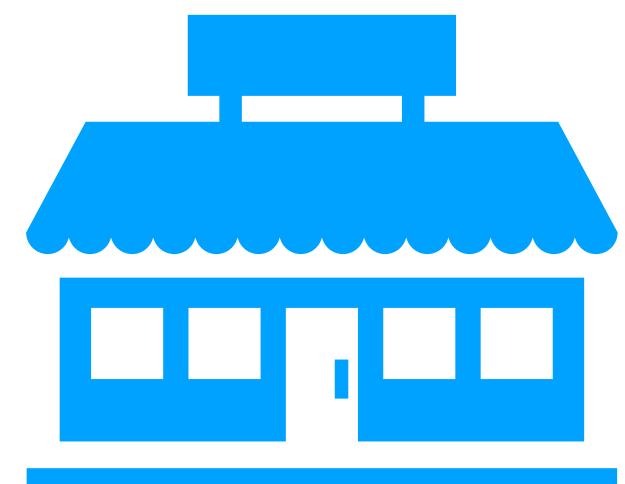


Web

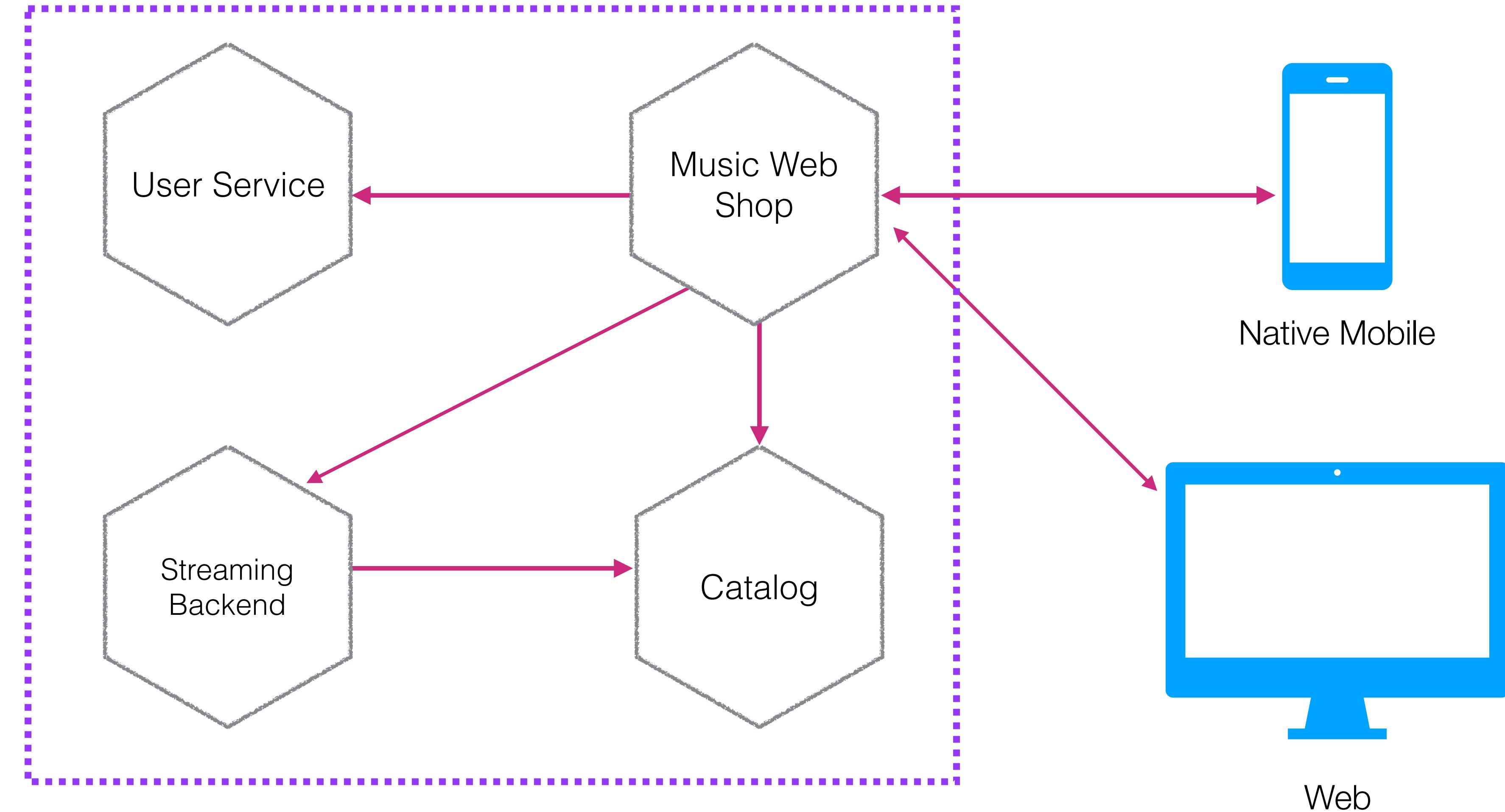
# MUSIC CORP 2018



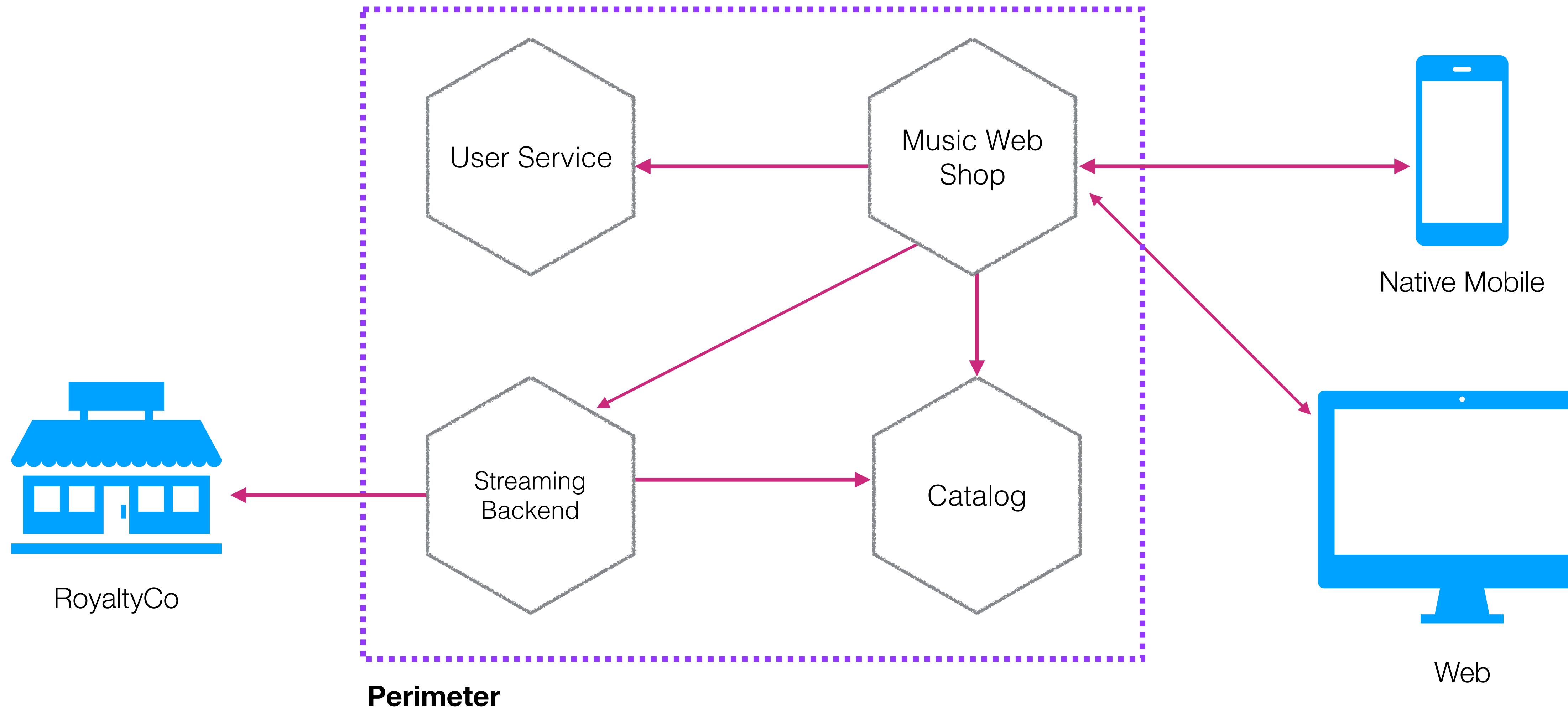
# MUSIC CORP 2018



RoyaltyCo



# MUSIC CORP 2018



# KEY CONCERNS OF TRANSPORT SECURITY

# KEY CONCERNS OF TRANSPORT SECURITY

Observation of data

## KEY CONCERNS OF TRANSPORT SECURITY

Observation of data

Manipulation of data

## KEY CONCERNS OF TRANSPORT SECURITY

Observation of data

Manipulation of data

Restricting access to endpoints

## KEY CONCERNS OF TRANSPORT SECURITY

Observation of data

Manipulation of data

Restricting access to endpoints

Impersonation of endpoints

# **HTTPS Everywhere!**

# HTTP + TLS

Server guarantees!

Server guarantees!

Payload not manipulated

Server guarantees!

Payload not manipulated

Client guarantees?

Server guarantees!

Payload not manipulated

Client guarantees?

Certificate management can be painful

## LET'S ENCRYPT



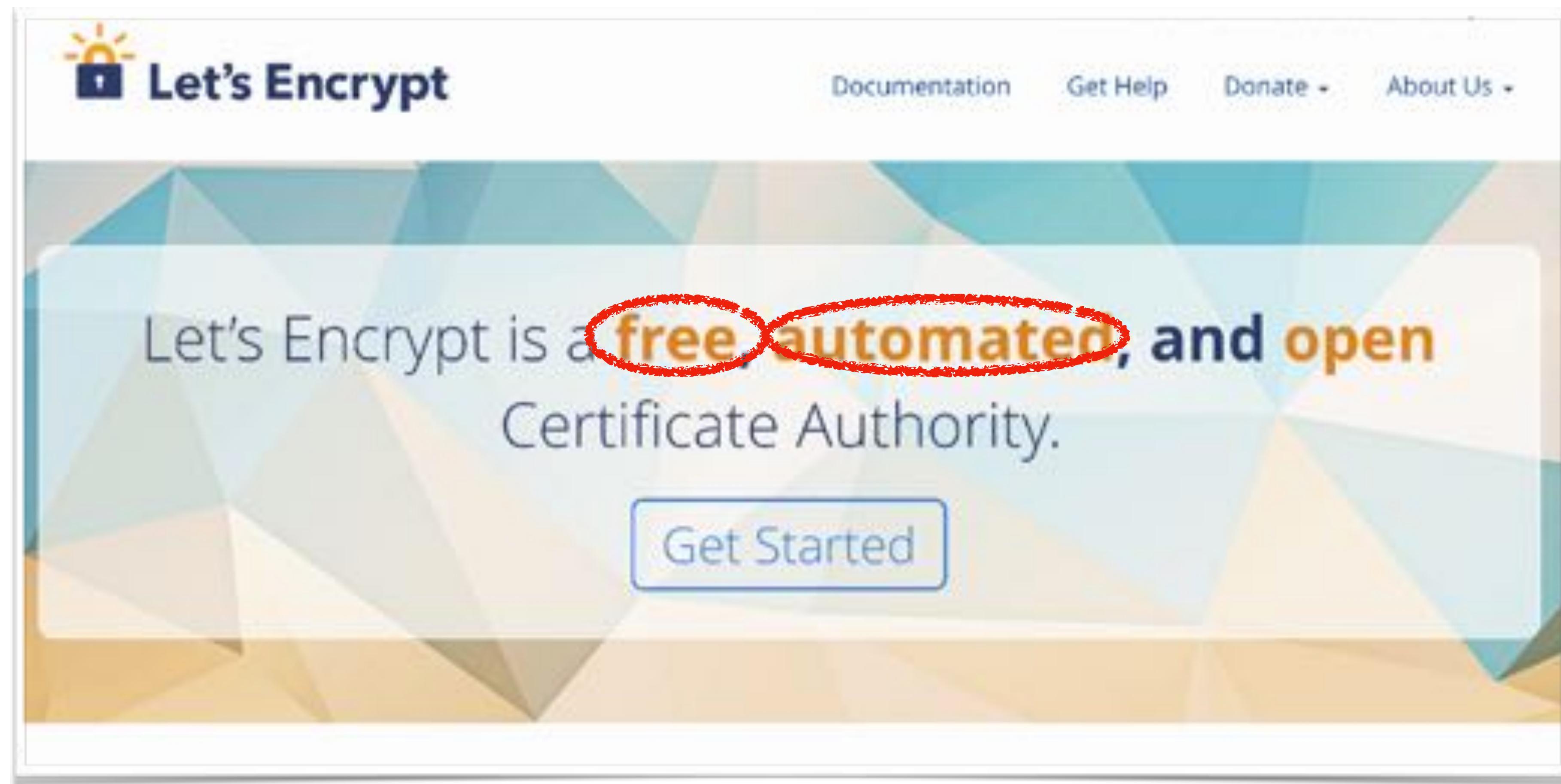
<https://letsencrypt.org/>

## LET'S ENCRYPT



<https://letsencrypt.org/>

## LET'S ENCRYPT



<https://letsencrypt.org/>

# AWS CERTIFICATE MANAGER

The screenshot shows the AWS Certificate Manager landing page. At the top, the title "AWS Certificate Manager" is displayed in orange. Below the title, a large paragraph explains the service's purpose: provisioning, managing, and deploying SSL/TLS certificates for AWS services. To the right, there is a call-to-action box with the text "Manage Your AWS Resources" and a "Sign in to the Console" button.

AWS Certificate Manager is a service that lets you easily provision, manage, and deploy Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services. SSL/TLS certificates are used to secure network communications and establish the identity of websites over the Internet. AWS Certificate Manager removes the time-consuming manual process of purchasing, uploading, and renewing SSL/TLS certificates. With AWS Certificate Manager, you can quickly request a certificate, deploy it on AWS resources such as Elastic Load Balancers, Amazon CloudFront distributions, and APIs on API Gateway, and let AWS Certificate Manager handle certificate renewals. SSL/TLS certificates provisioned through AWS Certificate Manager are free. You pay only for the AWS resources you create to run your application.

<https://aws.amazon.com/certificate-manager/>

## HOW DOES THIS STACK UP?

Server guarantees!

Payload not manipulated

Client guarantees?

Certificate management can be  
painful

## HOW DOES THIS STACK UP?

Server guarantees!

Payload not manipulated

Client guarantees?

Certificate management can be  
painful

Observation of data

Manipulation of data

Restricting access to endpoints

Impersonation of endpoints

## HOW DOES THIS STACK UP?

Server guarantees!



Observation of data

Payload not manipulated

Manipulation of data

Client guarantees?

Restricting access to endpoints

Certificate management can be  
painful

Impersonation of endpoints

## HOW DOES THIS STACK UP?

Server guarantees!

✓ Observation of data

Payload not manipulated

✓ Manipulation of data

Client guarantees?

Restricting access to endpoints

Certificate management can be  
painful

Impersonation of endpoints

## HOW DOES THIS STACK UP?

Server guarantees!

✓ Observation of data

Payload not manipulated

✓ Manipulation of data

Client guarantees?

? Restricting access to endpoints

Certificate management can be  
painful

Impersonation of endpoints

## HOW DOES THIS STACK UP?

Server guarantees!

✓ Observation of data

Payload not manipulated

✓ Manipulation of data

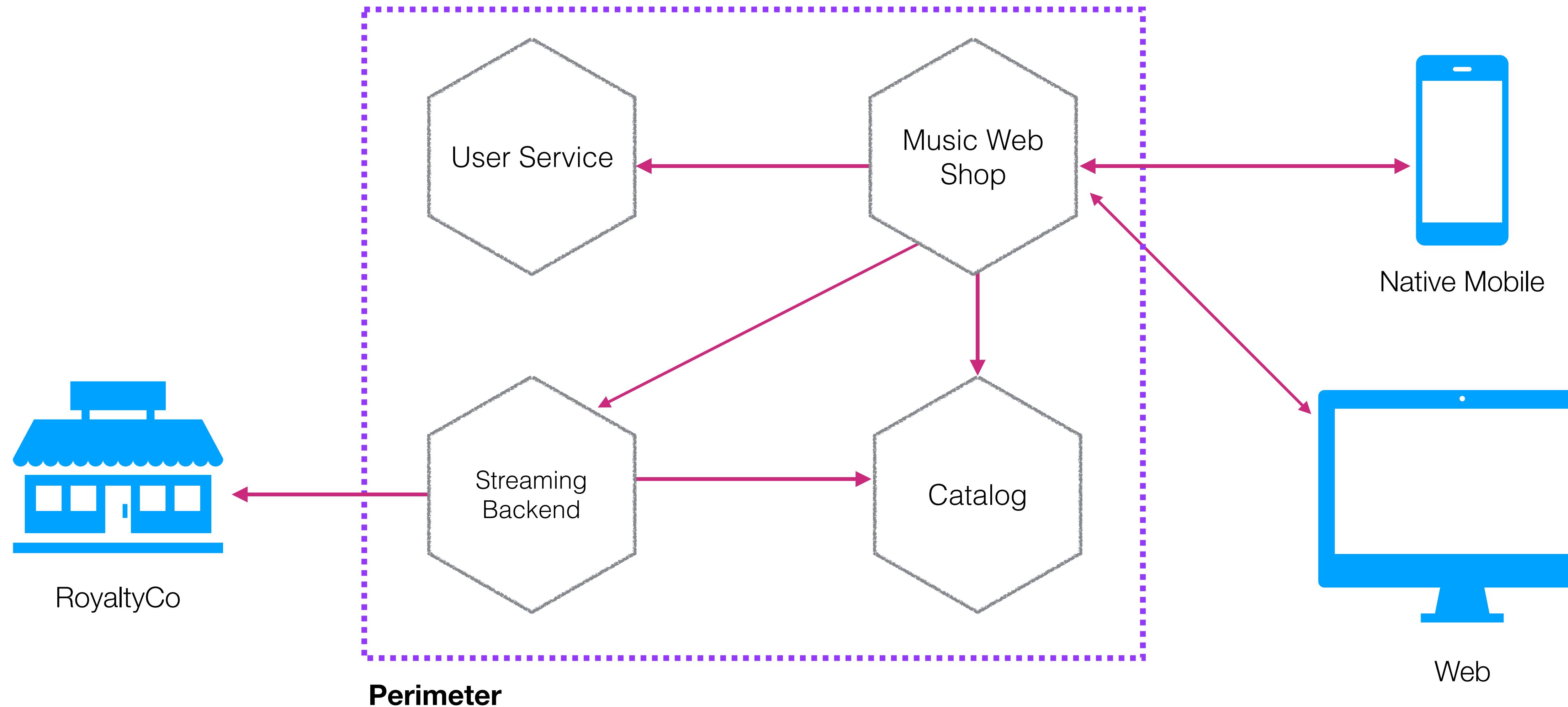
Client guarantees?

? Restricting access to endpoints

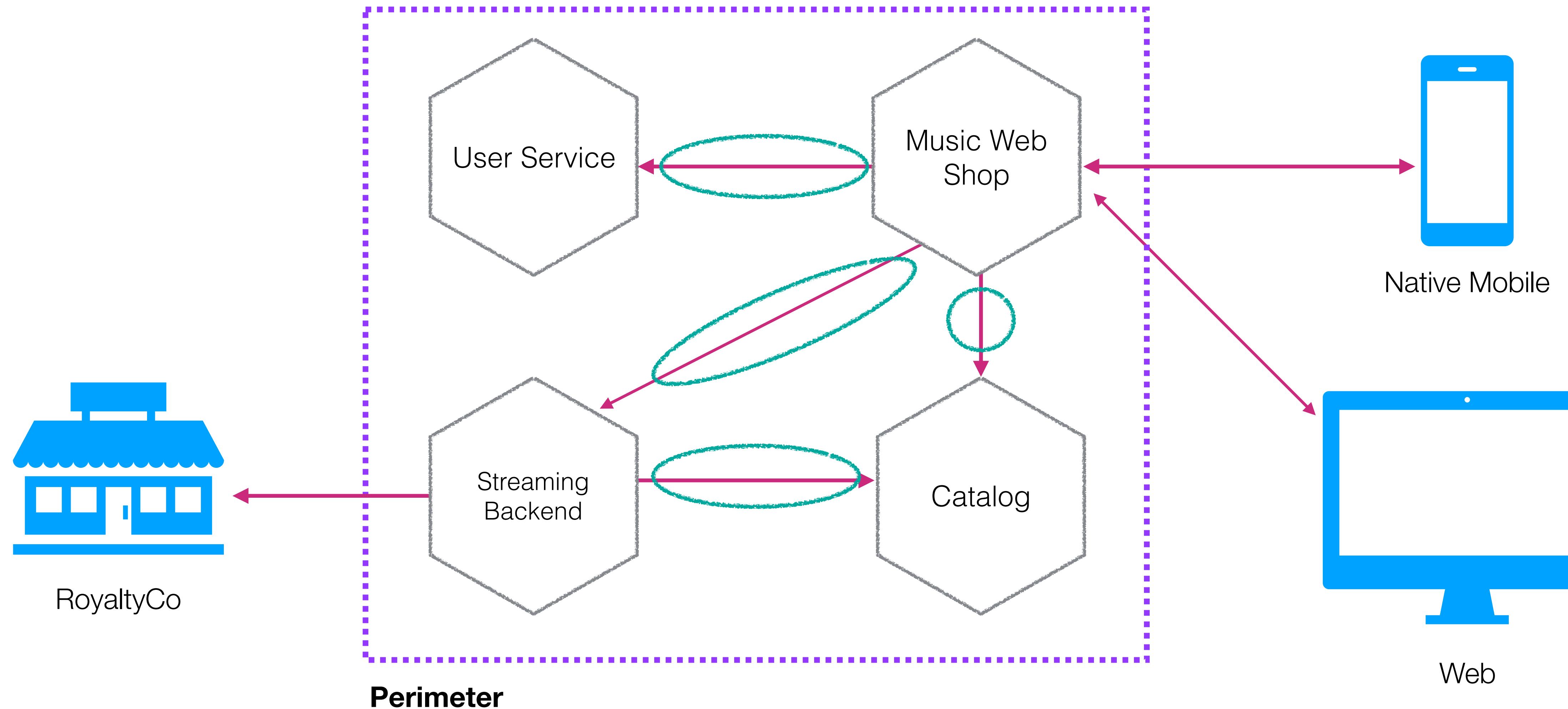
Certificate management can be  
painful

✓ Impersonation of endpoints

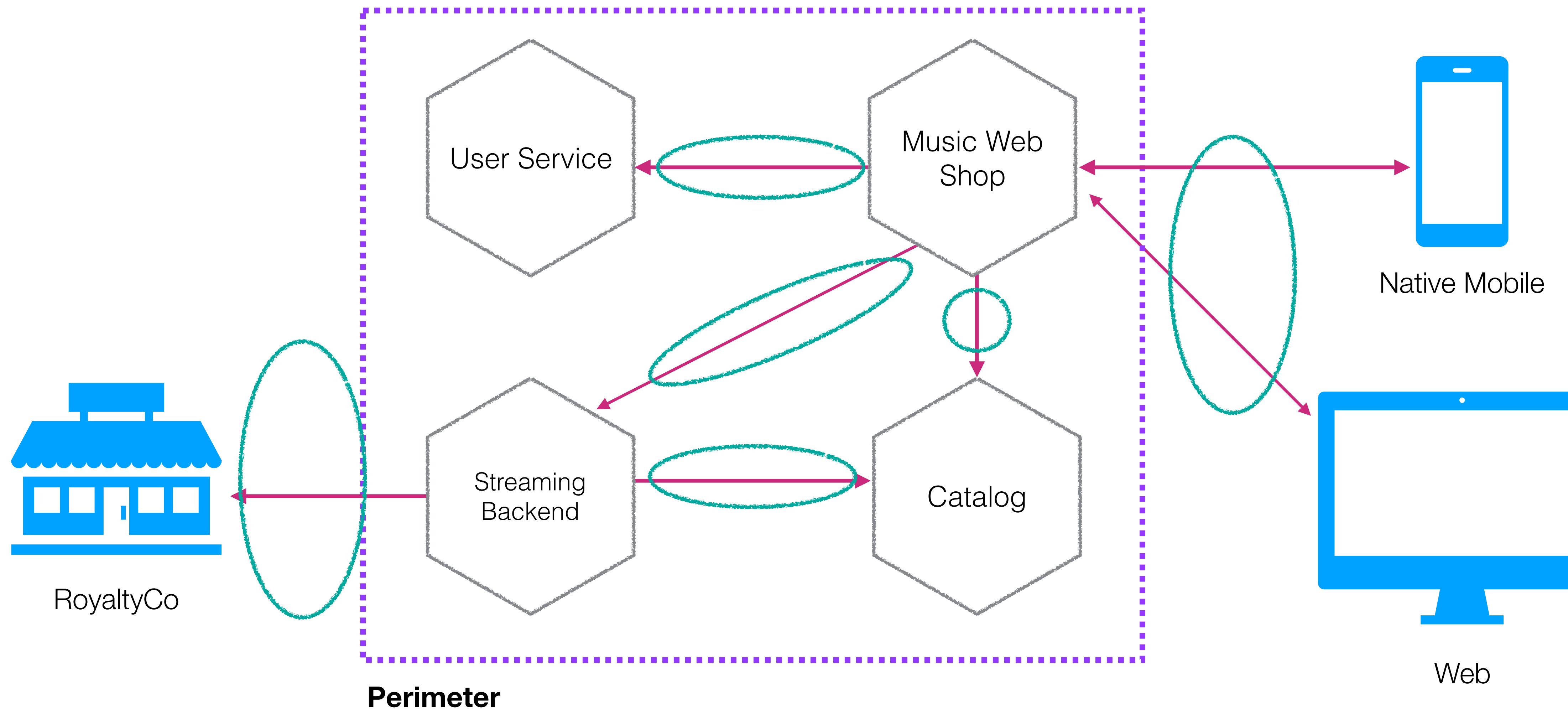
# HTTPS EVERYWHERE!



# HTTPS EVERYWHERE!

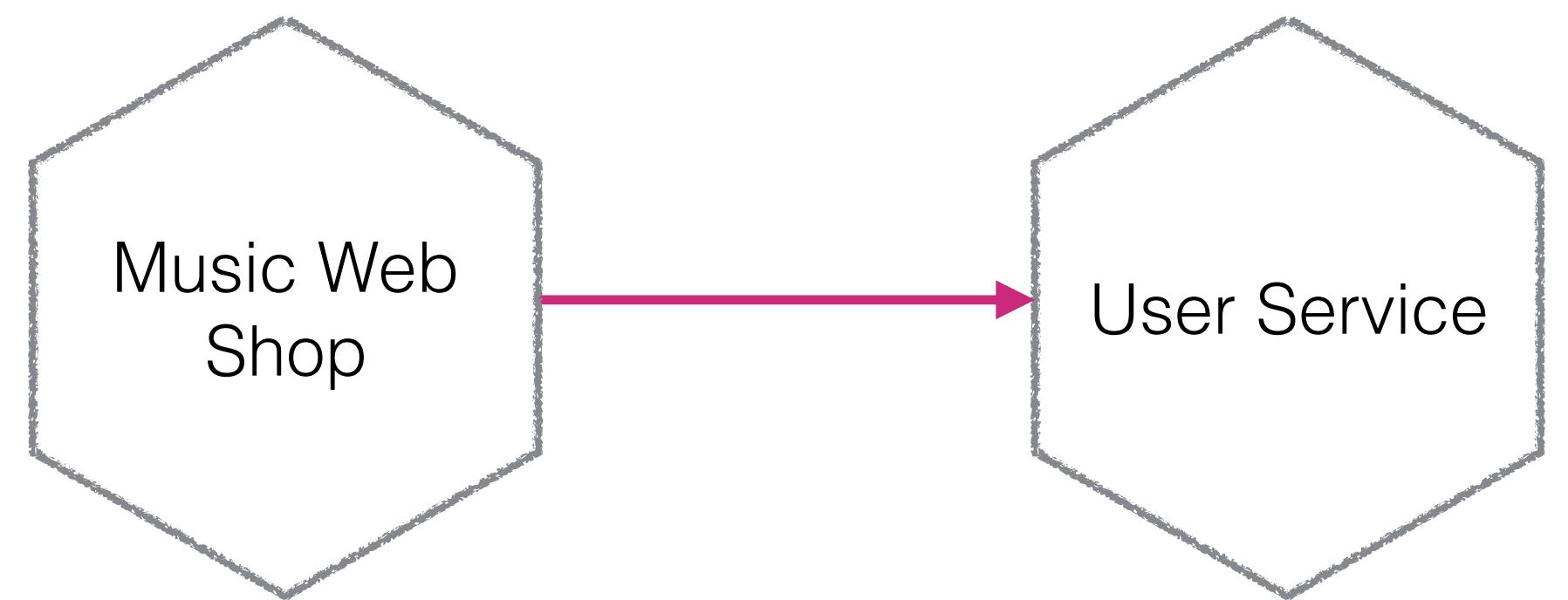


# HTTPS EVERYWHERE!

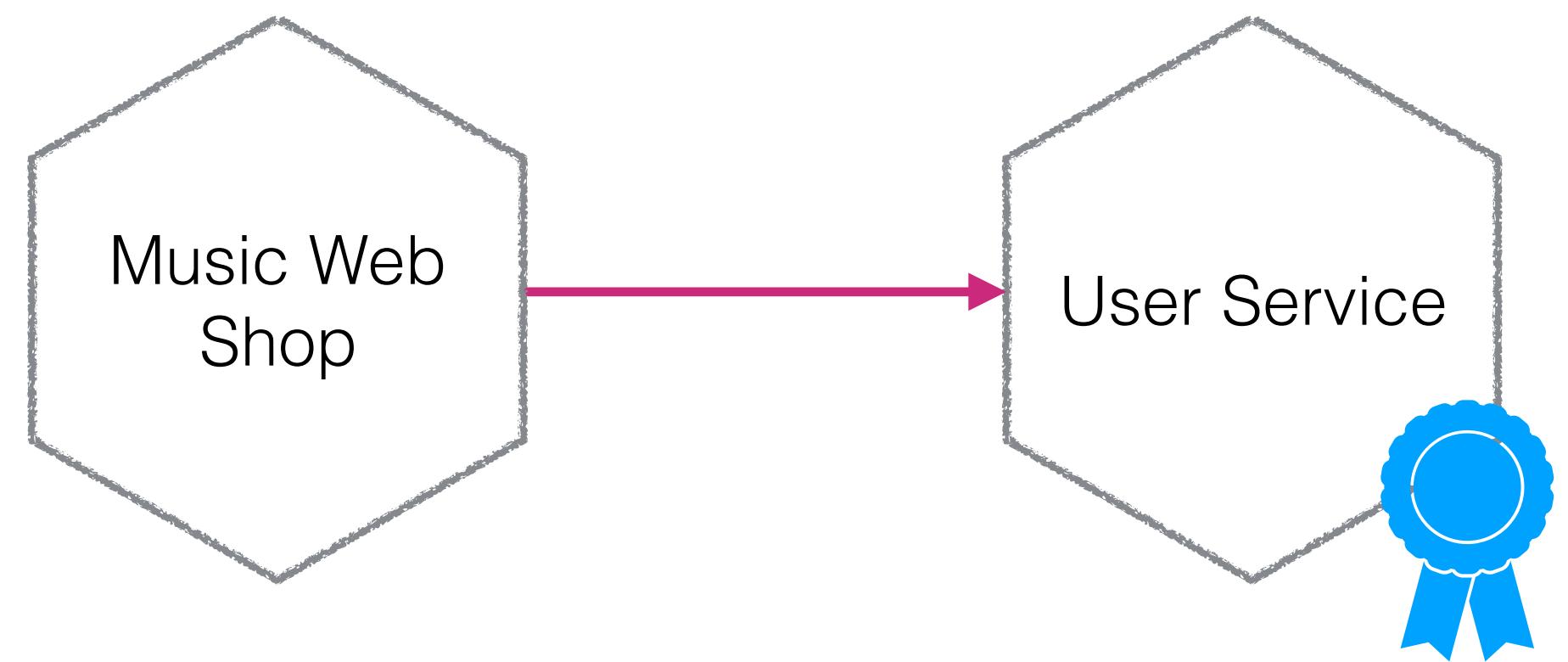


# Mutual TLS

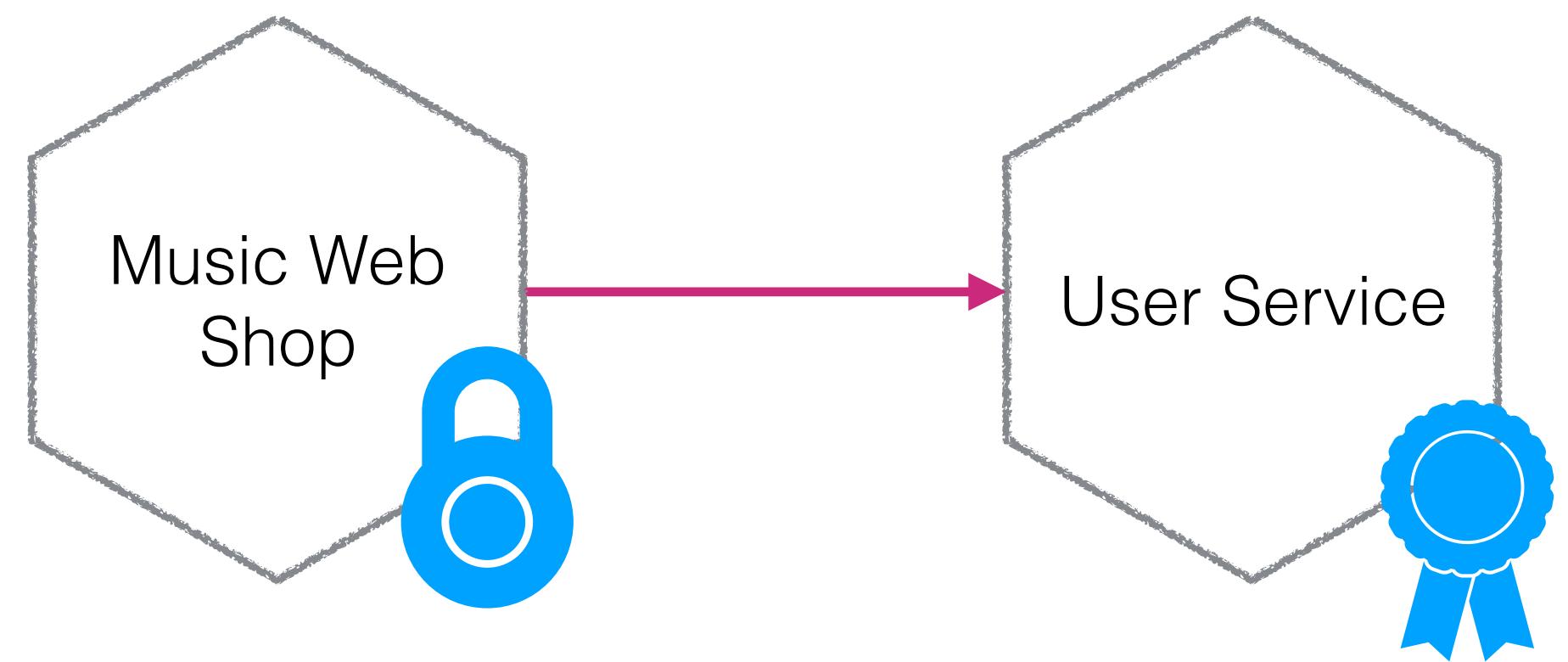
# MUTUAL TLS



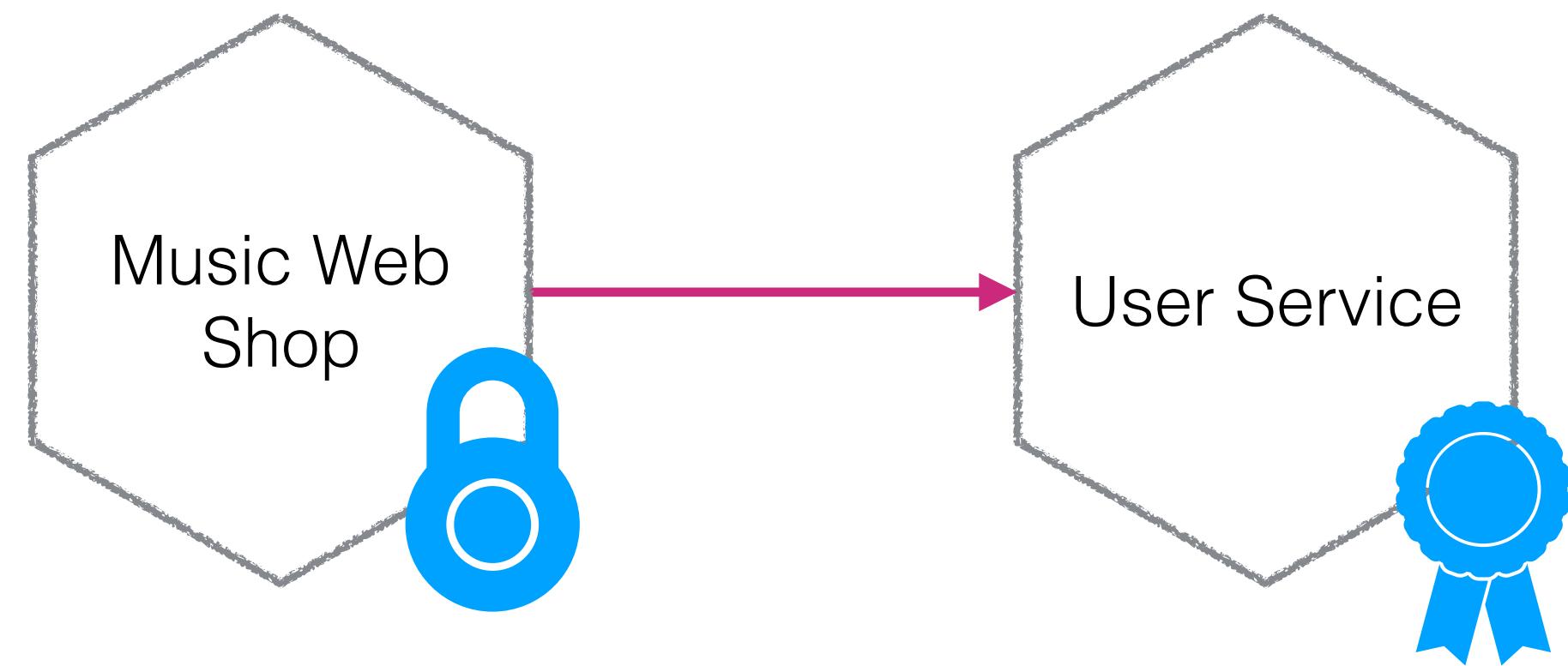
# MUTUAL TLS



# MUTUAL TLS

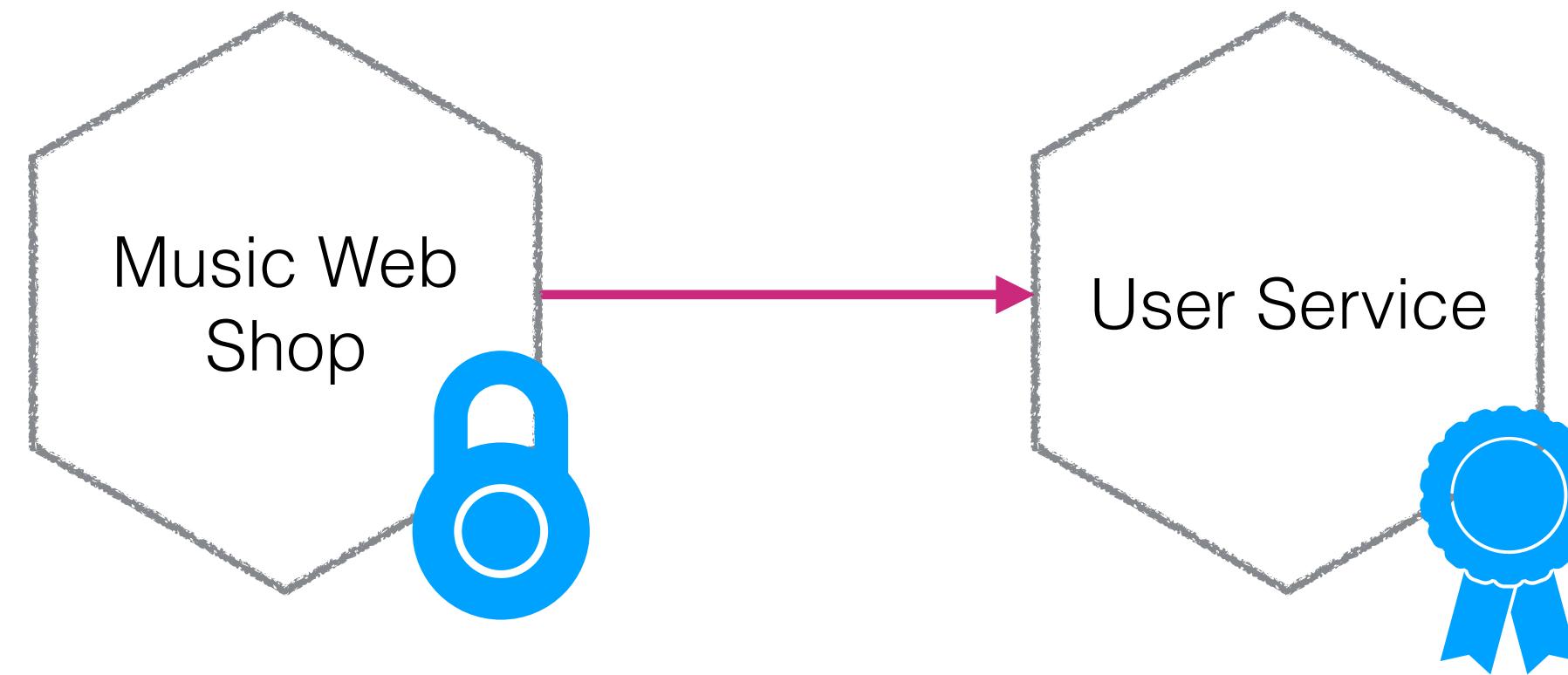


## MUTUAL TLS



Client and server guarantees!

## MUTUAL TLS



Client and server guarantees!

Certificate management is REALLY painful

## AZURE - CLIENT-SIDE CERTIFICATE MANAGEMENT

# How to secure back-end services using client certificate authentication in Azure API Management

10/30/2017 • 3 minutes to read • Contributors  [all](#)

## In this article

[Prerequisites](#)

[Upload a client certificate](#)

[Delete a client certificate](#)

[Configure an API to use a client certificate for gateway authentication](#)

[Self-signed certificates](#)

[Next steps](#)

API Management provides the capability to secure access to the back-end service of an API using client certificates. This guide shows how to manage certificates in the API publisher portal, and how to configure an API to use a certificate to access its back-end service.

For information about managing certificates using the API Management REST API, see [Azure API Management REST API Certificate entity](#).

<https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-mutual-certificates>

# AWS - CLIENT-SIDE CERTIFICATE MANAGEMENT

AWS Documentation » Amazon API Gateway » Developer Guide » Controlling Access to an API in API Gateway » Use Client-Side SSL Certificates for Authentication by the Backend

## Use Client-Side SSL Certificates for Authentication by the Backend

You can use API Gateway to generate an SSL certificate and use its public key in the backend to verify that HTTP requests to your backend system are from API Gateway. This allows your HTTP backend to control and accept only requests originating from Amazon API Gateway, even if the backend is publicly accessible.

### Note

Some backend servers may not support SSL client authentication as API Gateway does and could return an SSL certificate error. For a list of incompatible backend servers, see [Known Issues](#).

The SSL certificates that are generated by API Gateway are self-signed and only the public key of a certificate is visible in the API Gateway console or through the APIs.

### Topics

- [Generate a Client Certificate Using the API Gateway Console](#)
- [Configure an API to Use SSL Certificates](#)
- [Test Invoke](#)
- [Configure Backend to Authenticate API](#)

<https://docs.aws.amazon.com/apigateway/latest/developerguide/getting-started-client-side-ssl-authentication.html>

## MUTUAL TLS

Observation of data

Manipulation of data

Restricting access to endpoints

Impersonation of endpoints

## MUTUAL TLS

✓ Observation of data

Manipulation of data

Restricting access to endpoints

Impersonation of endpoints

## MUTUAL TLS

- ✓ Observation of data
- ✓ Manipulation of data

Restricting access to endpoints

Impersonation of endpoints

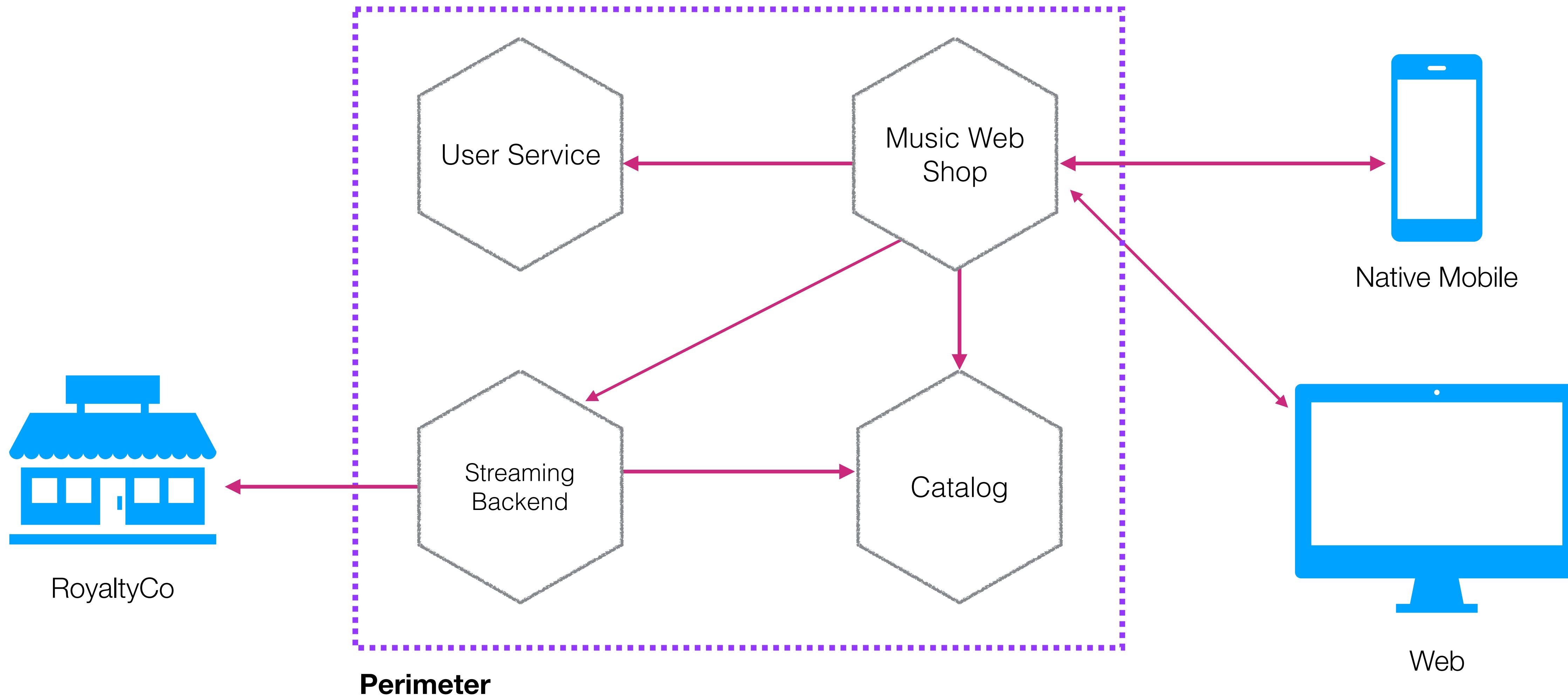
## MUTUAL TLS

- ✓ Observation of data
  - ✓ Manipulation of data
  - ✓ Restricting access to endpoints
- Impersonation of endpoints

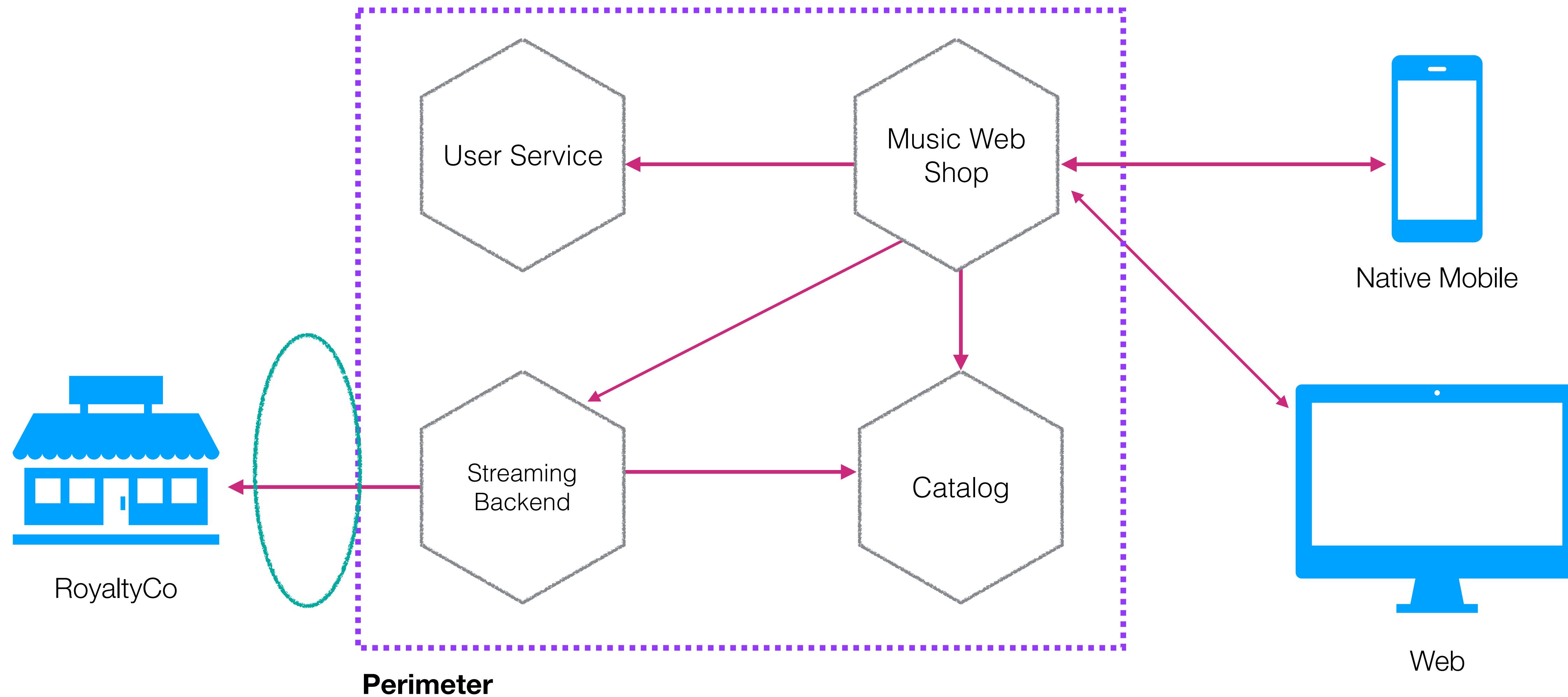
## MUTUAL TLS

- ✓ Observation of data
- ✓ Manipulation of data
- ✓ Restricting access to endpoints
- ✓ Impersonation of endpoints

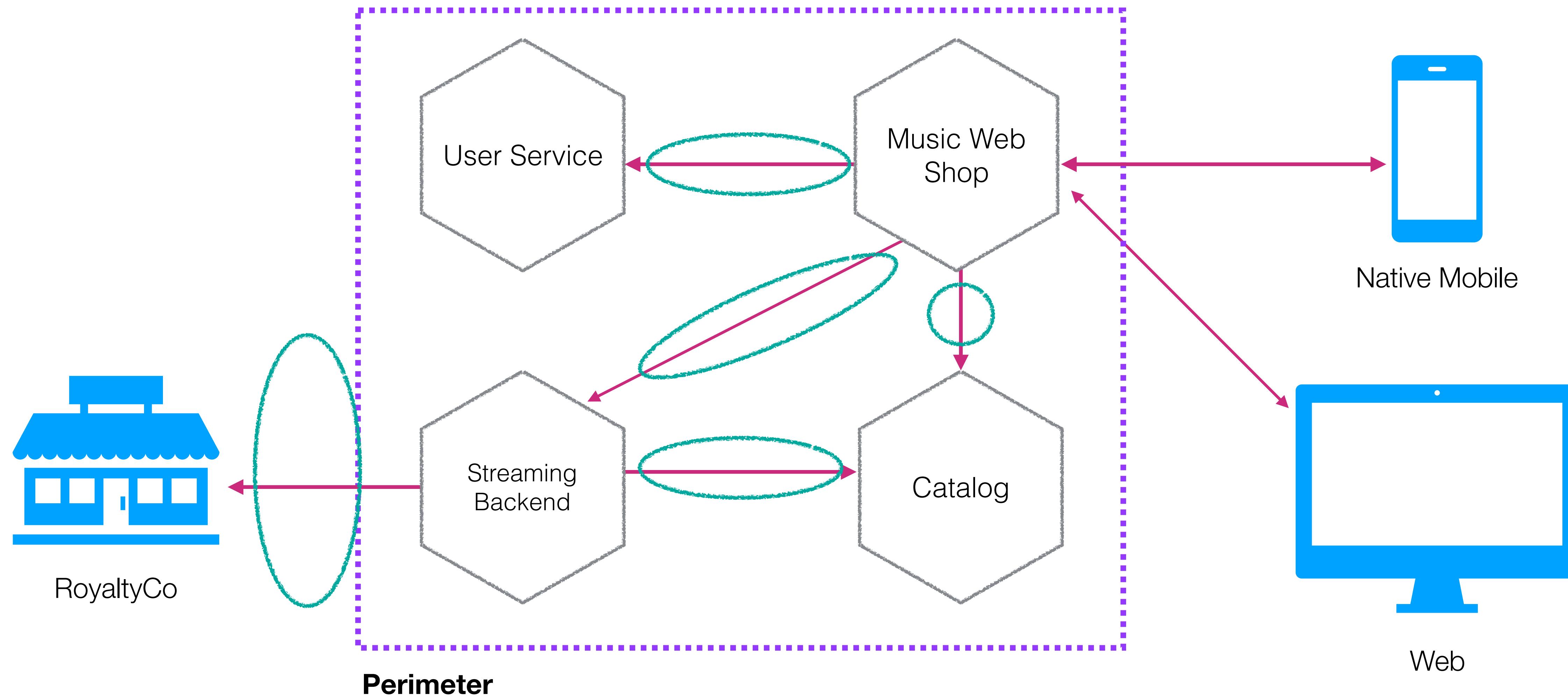
## MUTUAL TLS



## MUTUAL TLS



## MUTUAL TLS



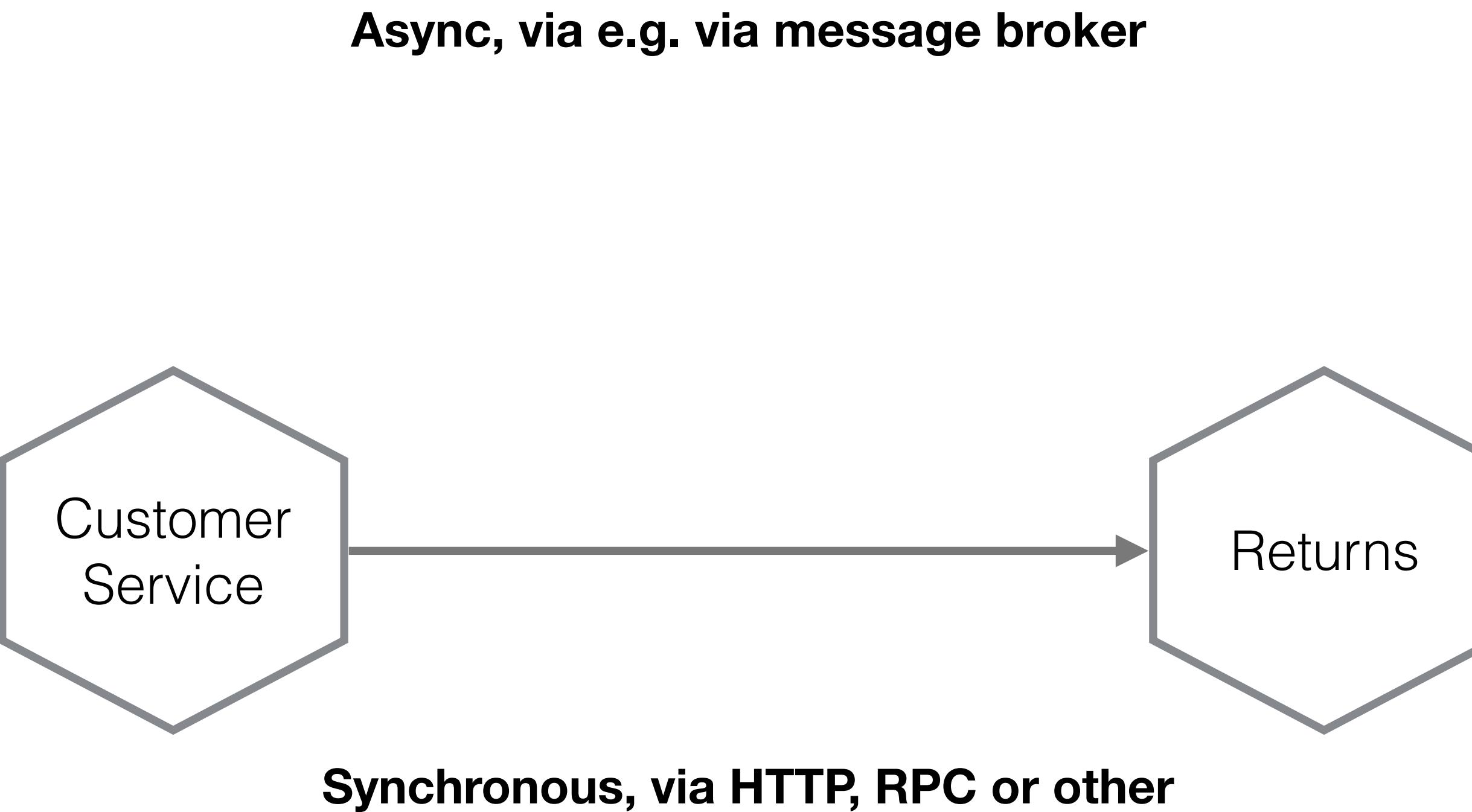
## OTHER PROTOCOLS?



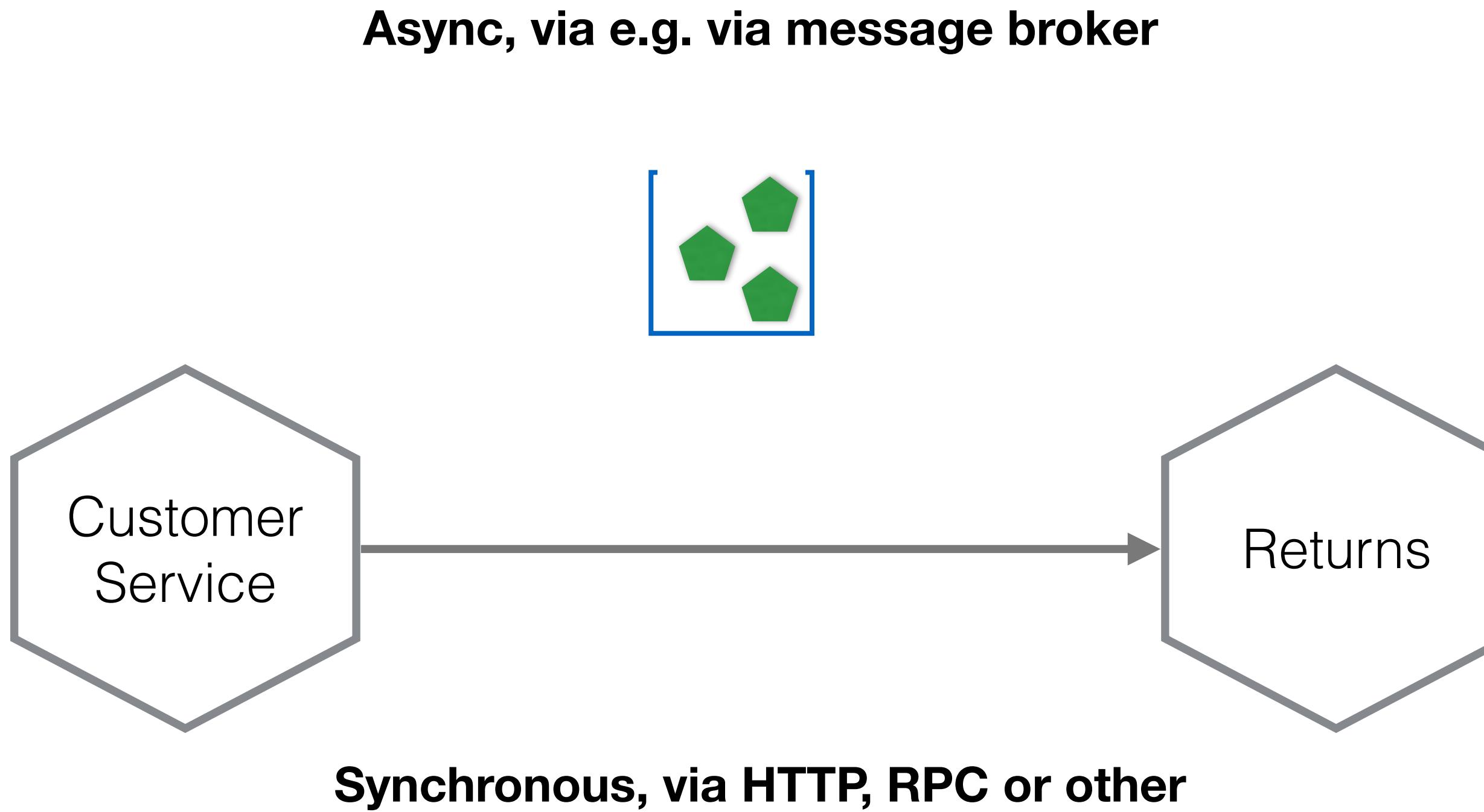
## OTHER PROTOCOLS?



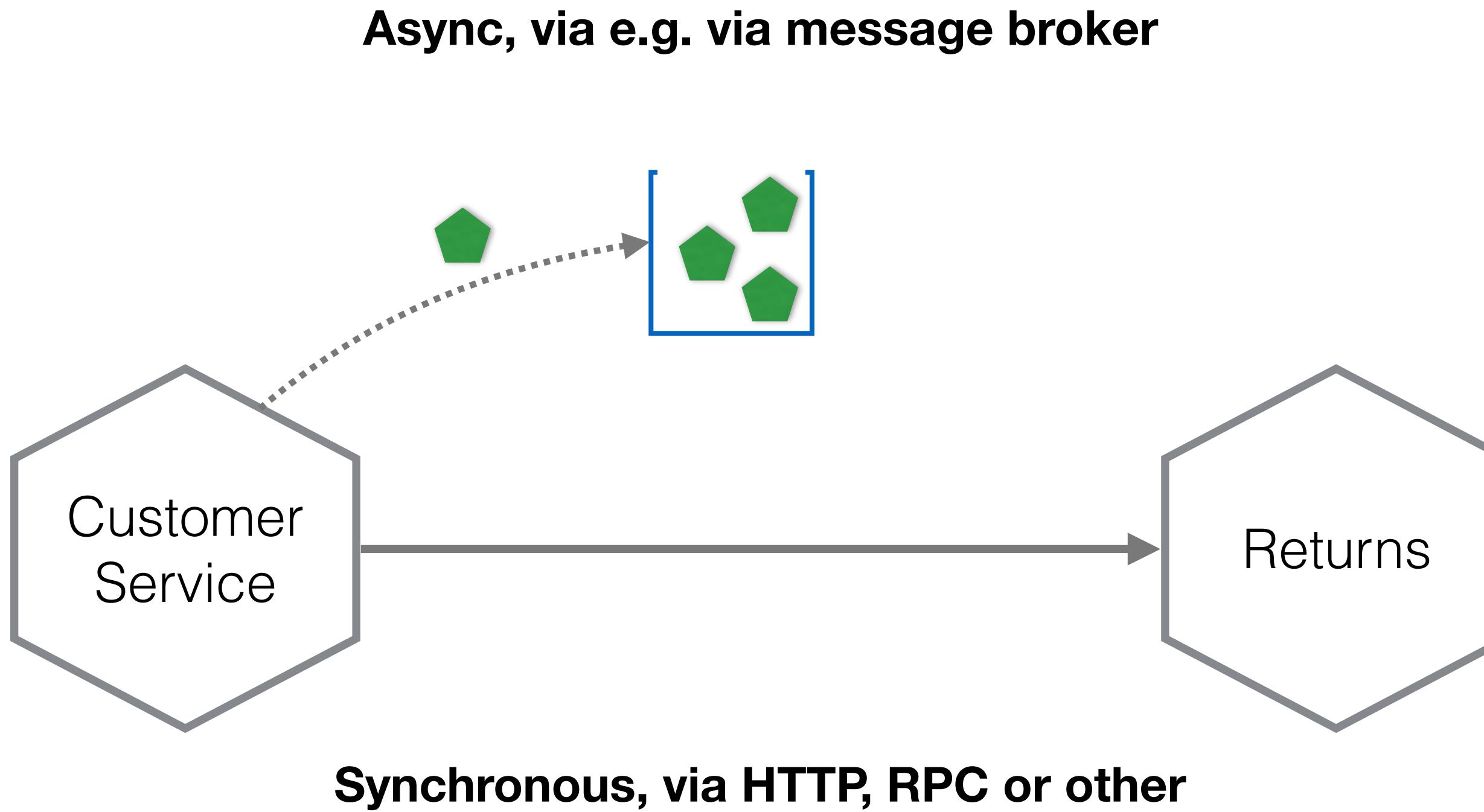
## OTHER PROTOCOLS?



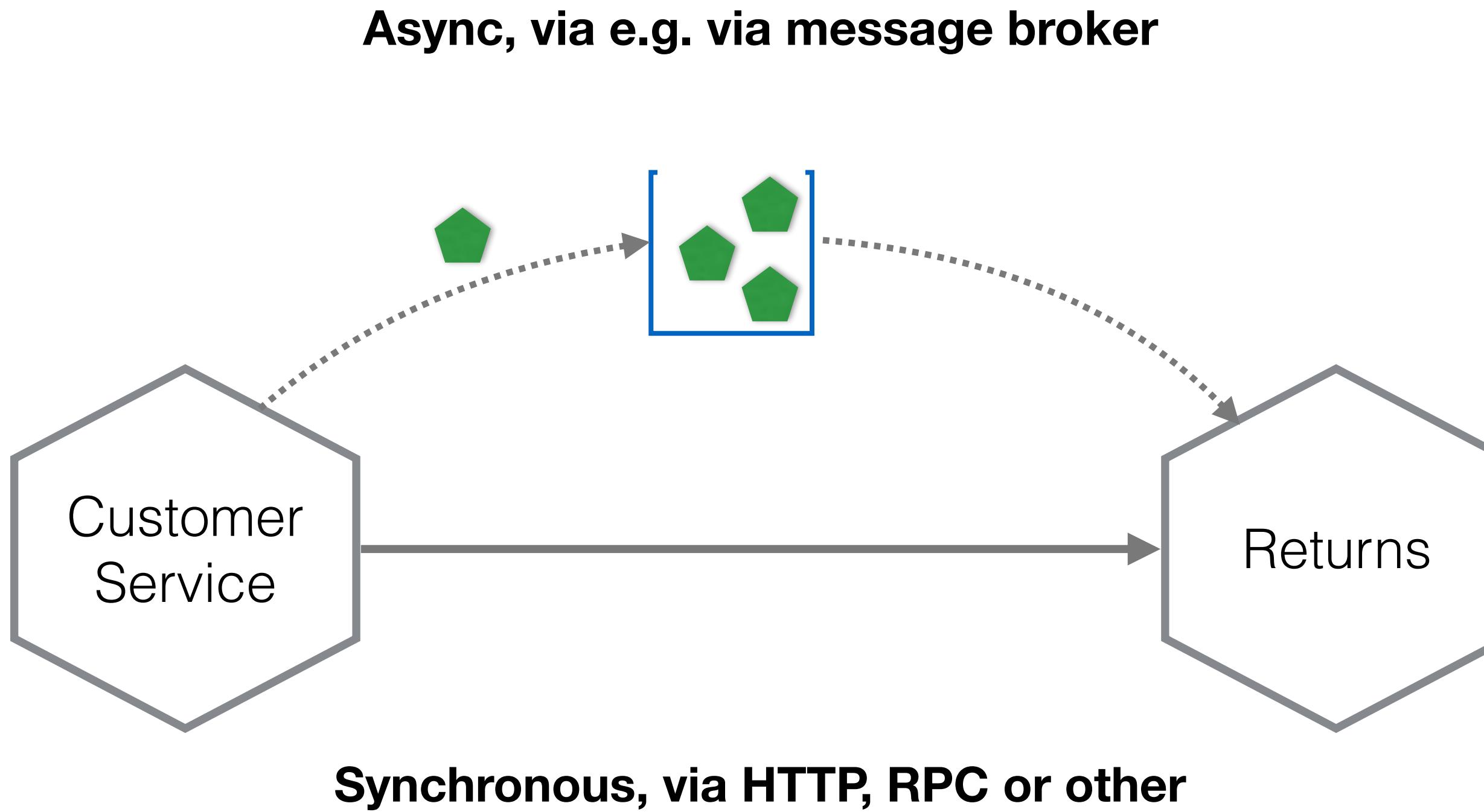
## OTHER PROTOCOLS?



## OTHER PROTOCOLS?



## OTHER PROTOCOLS?



## TLS Support

### Intro

RabbitMQ has inbuilt support for TLS. This includes client connections and popular plugins, where applicable, such as [Federation links](#). It is also possible to use TLS to [encrypt inter-node connections in clusters](#).

This guide covers various topics related to TLS in RabbitMQ:

[Enabling TLS listeners in RabbitMQ](#)

[How to generate self-signed certificates for development and QA environments](#)

[TLS configuration in Java and .NET clients](#)

[Known vulnerabilities and their migration](#)

[TLS version and cipher suite configuration](#)

[Certificate chain validation depth](#)

[Tools that can be used to evaluate a TLS setup](#)

and more. It is not, however, a primer on TLS, encryption, [Public Key Infrastructure](#) and related topics, so the concepts are covered very briefly. A number of beginner-oriented primers are available elsewhere on the Web: [one](#) [two](#), [three](#), [four](#).

## TLS Support

### Intro

RabbitMQ has inbuilt support for TLS. This includes client connections and popular plugins, where applicable, such as [Federation links](#). It is also possible to use TLS to [encrypt inter-node connections in clusters](#).

This guide covers various topics related to TLS in RabbitMQ:

[Enabling TLS listeners in RabbitMQ](#)

[How to generate self-signed certificates for development and QA environments](#)

[TLS configuration in Java and .NET clients](#)

[Known vulnerabilities and their migration](#)

[TLS version and cipher suite configuration](#)

[Certificate chain validation depth](#)

Tools th

[Erlang/OTP Requirements for TLS Support](#)

and more

topics, se

available

In order to support TLS connections, RabbitMQ needs TLS and crypto-related modules to be available in the Erlang/OTP installation. The recommended Erlang/OTP version to use with TLS is the most recent [supported Erlang release](#). Earlier versions, even if they are supported, may work for most certificates but have known limitations (see below).

## TLS Support

### Intro

RabbitMQ has inbuilt support for TLS. This includes client connections and popular plugins, where applicable, such as [Federation links](#). It is also possible to use TLS to [encrypt inter-node connections in clusters](#).

This guide covers various topics related to TLS in RabbitMQ:

[Enabling TLS listeners in RabbitMQ](#)

[How to generate self-signed certificates for development and QA environments](#)

[TLS configuration in Java and .NET clients](#)

[Known vulnerabilities and their migration](#)

[TLS version and cipher suite configuration](#)

[Certificate chain validation depth](#)

Tools th

[Erlang/OTP Requirements for TLS Support](#)

and more

topics, se

available

In order to support TLS connections, RabbitMQ needs TLS and crypto-related modules to be available in the Erlang/OTP installation. The recommended Erlang/OTP version to use with TLS is the most recent [supported Erlang release](#). Earlier versions, even if they are supported, may work for most certificates but have known limitations (see below).

## Access Control (Authentication, Authorisation) in RabbitMQ

This document describes authentication and authorisation machinery that implements access control. Authentication backends should not be confused with [authentication mechanisms](#) in AMQP 0-9-1.

A separate guide covers multiple topics around [passwords](#). It is only applicable to the internal authentication backend.

### Terminology and Definitions

Authentication and authorisation are often confused or used interchangeably. That's wrong and in RabbitMQ, the two are separated. For the sake of simplicity, we'll define authentication as "identifying who the user is" and authorisation as "determining what the user is and isn't allowed to do."

### Default Virtual Host and User

When the server first starts running, and detects that its database is uninitialised or has been deleted, it initialises a fresh database with the following resources:

## TLS, Kerberos, SASL, and Authorizer in Apache Kafka 0.9 – Enabling New Encryption, Authorization, and Authentication Features

Apache Kafka is frequently used to store critical data making it one of the most important components of a company's data infrastructure. Our goal is to make it possible to run Kafka as a central platform for streaming data, supporting anything from a single app to a whole company. Multi-tenancy is an essential requirement in achieving this vision and, in turn, security features are crucial for multi-tenancy.

Previous to 0.9, Kafka had no built-in security features. One could lock down access at the network level but this is not viable for a big shared multi-tenant cluster being used across a large company. Consequently securing Kafka has been one of the most requested features. Security is of particular importance in today's world where cyber-attacks are a common occurrence and the threat of data breaches is a reality for businesses of all sizes, and at all levels from individual users to whole government entities.

Four key security features were added in Apache Kafka 0.9, which is included in the Confluent Platform 2.0:

## TLS, Kerberos, SASL, and Authorizer in Apache Kafka 0.9 – Enabling New Encryption, Authorization, and Authentication Features

Apache Kafka is frequently used to store company's data infrastructure. Our goal is supporting anything from a single app to this vision and, in turn, security features are

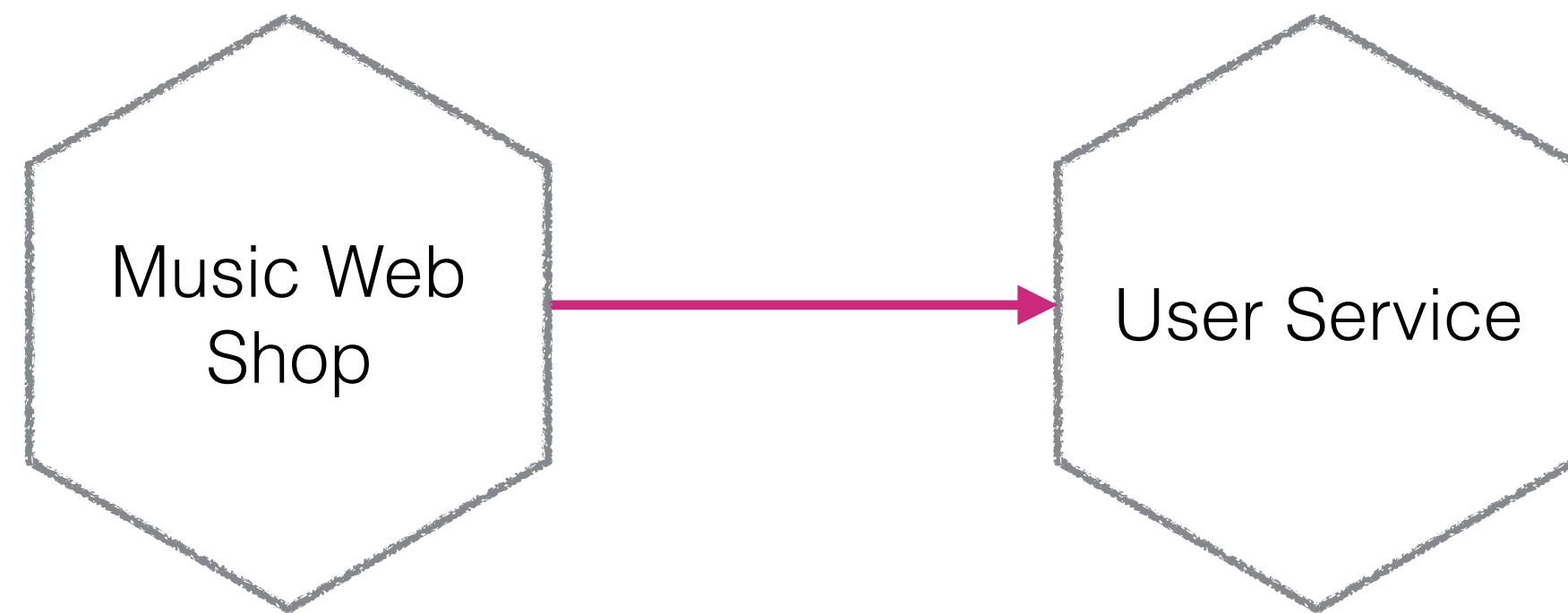
Previous to 0.9, Kafka had no built-in security, so it was not viable for a big shared multi-tenant cluster. This has been one of the most requested features because data attacks are a common occurrence and the security needs range across levels from individual users to whole government entities.

Four key security features were added in [Apache Kafka 0.9](#), which is included in the [Confluent Platform 2.0](#):

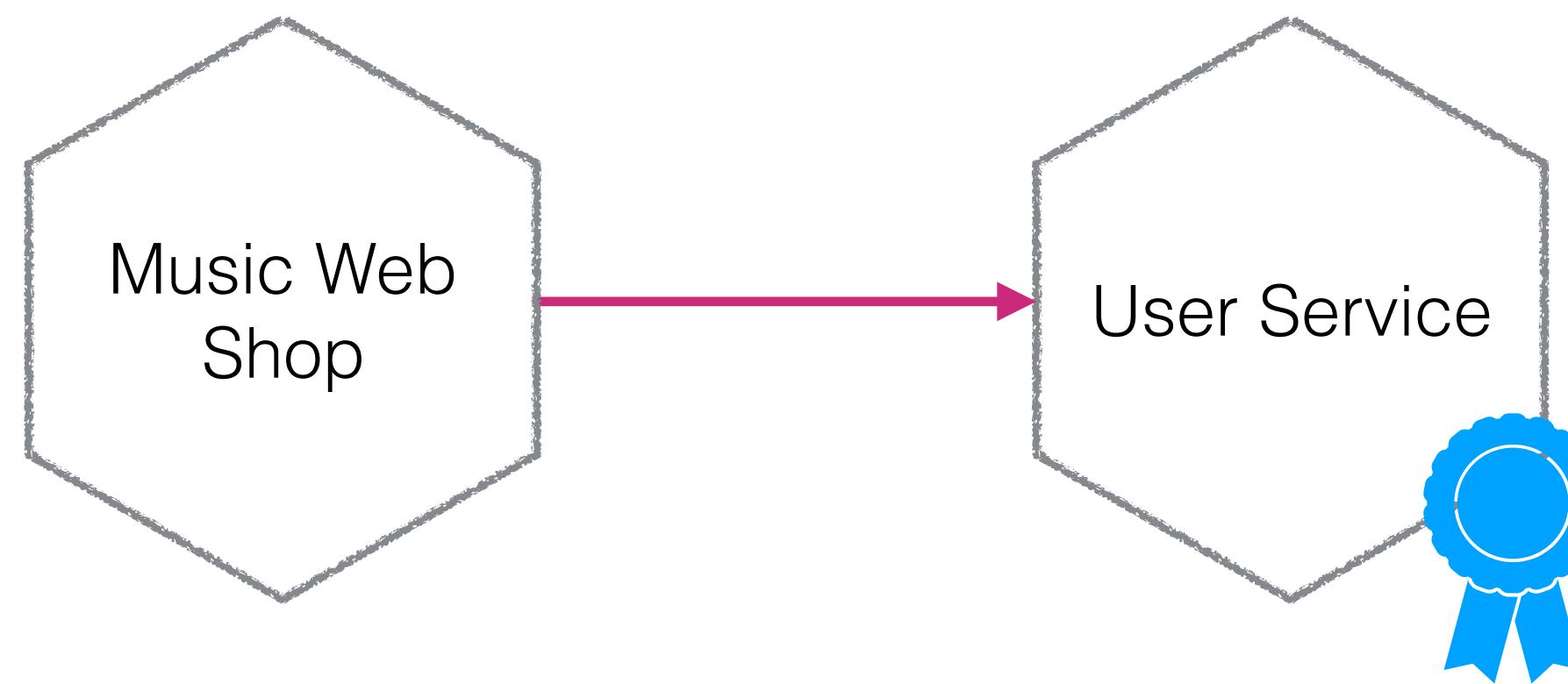
1. Administrators can require client authentication using either Kerberos or Transport Layer Security (TLS) client certificates, so that Kafka brokers know who is making each request.
2. A Unix-like permissions system can be used to control which users can access which data.
3. Network communication can be encrypted, allowing messages to be securely sent across untrusted networks.
4. Administrators can require authentication for communication between Kafka brokers and ZooKeeper.

Four key security features were added in [Apache Kafka 0.9](#), which is included in the [Confluent Platform 2.0](#):

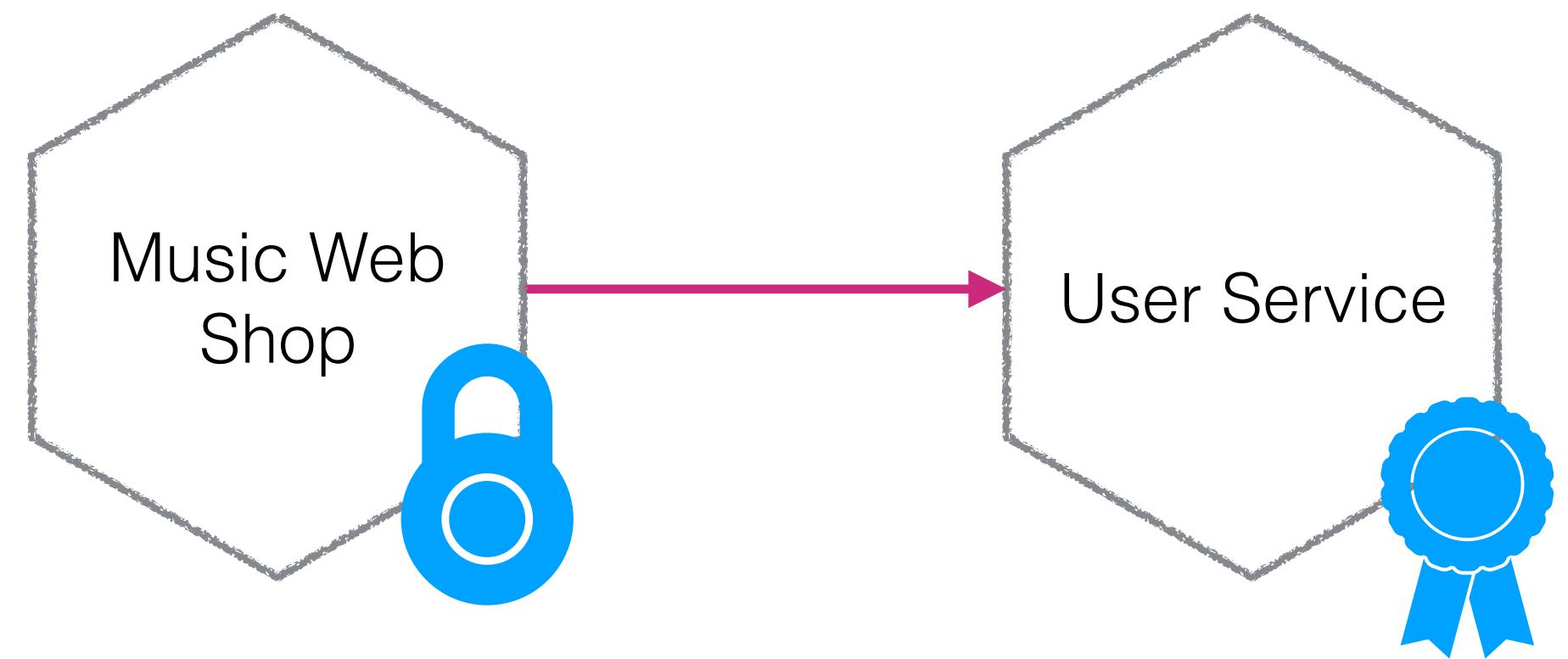
# TRANSPORT AUTHENTICATION



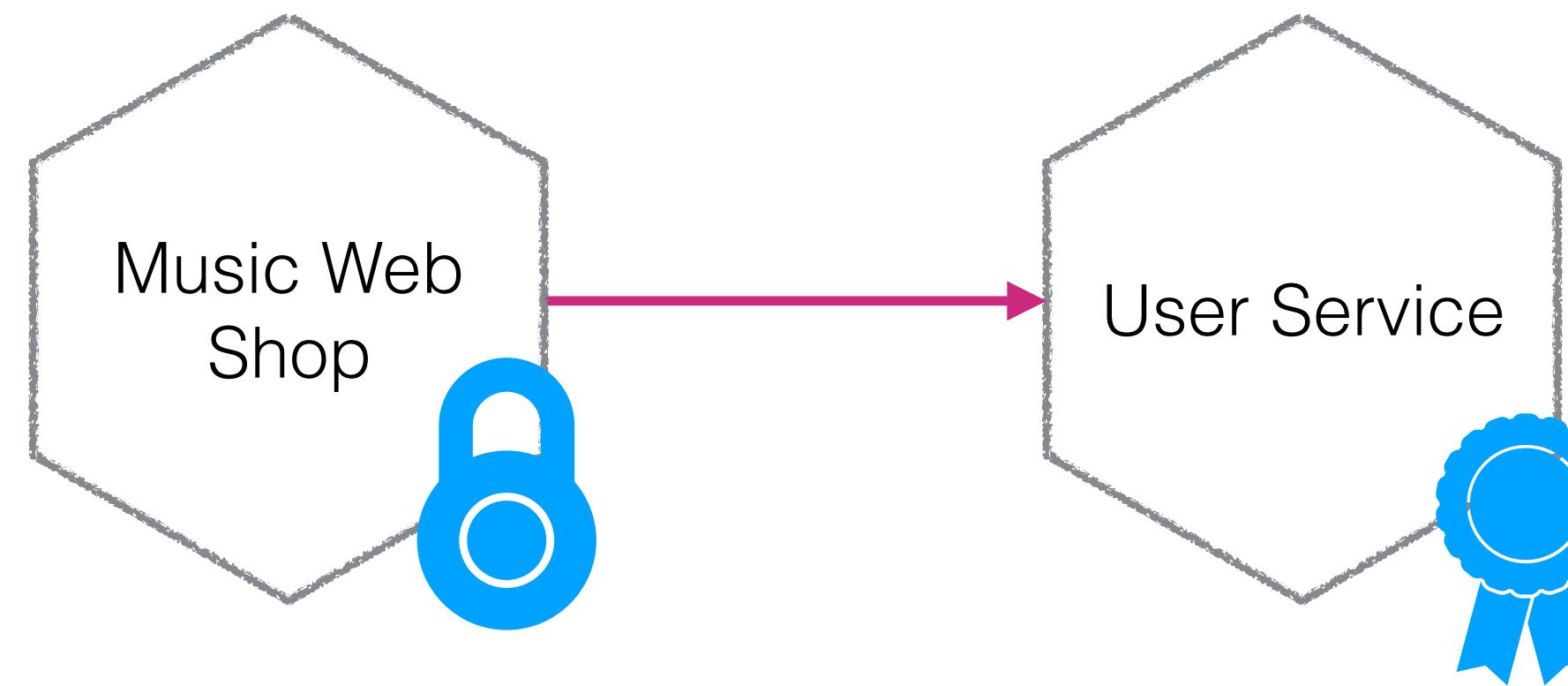
# TRANSPORT AUTHENTICATION



# TRANSPORT AUTHENTICATION

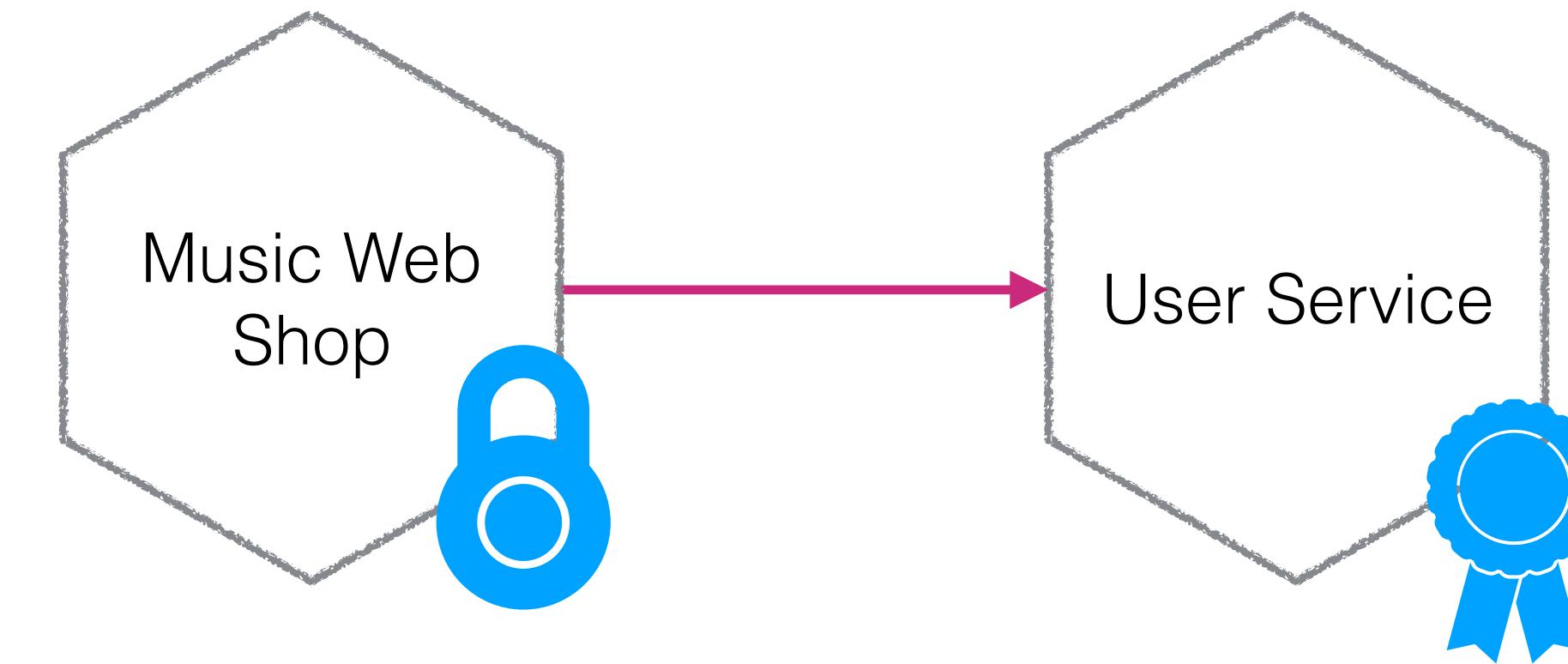


## TRANSPORT AUTHENTICATION



Server-side identity

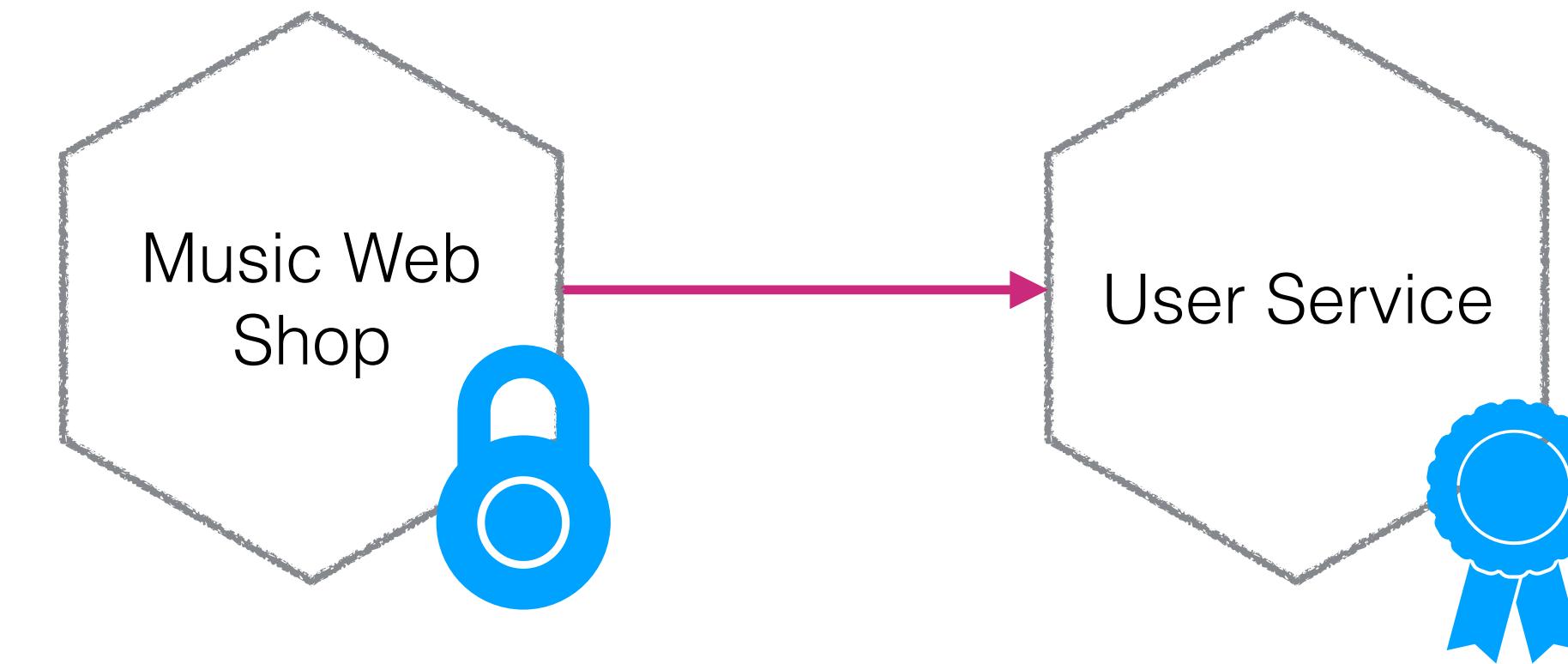
# TRANSPORT AUTHENTICATION



Client-side identity

Server-side identity

## TRANSPORT AUTHENTICATION



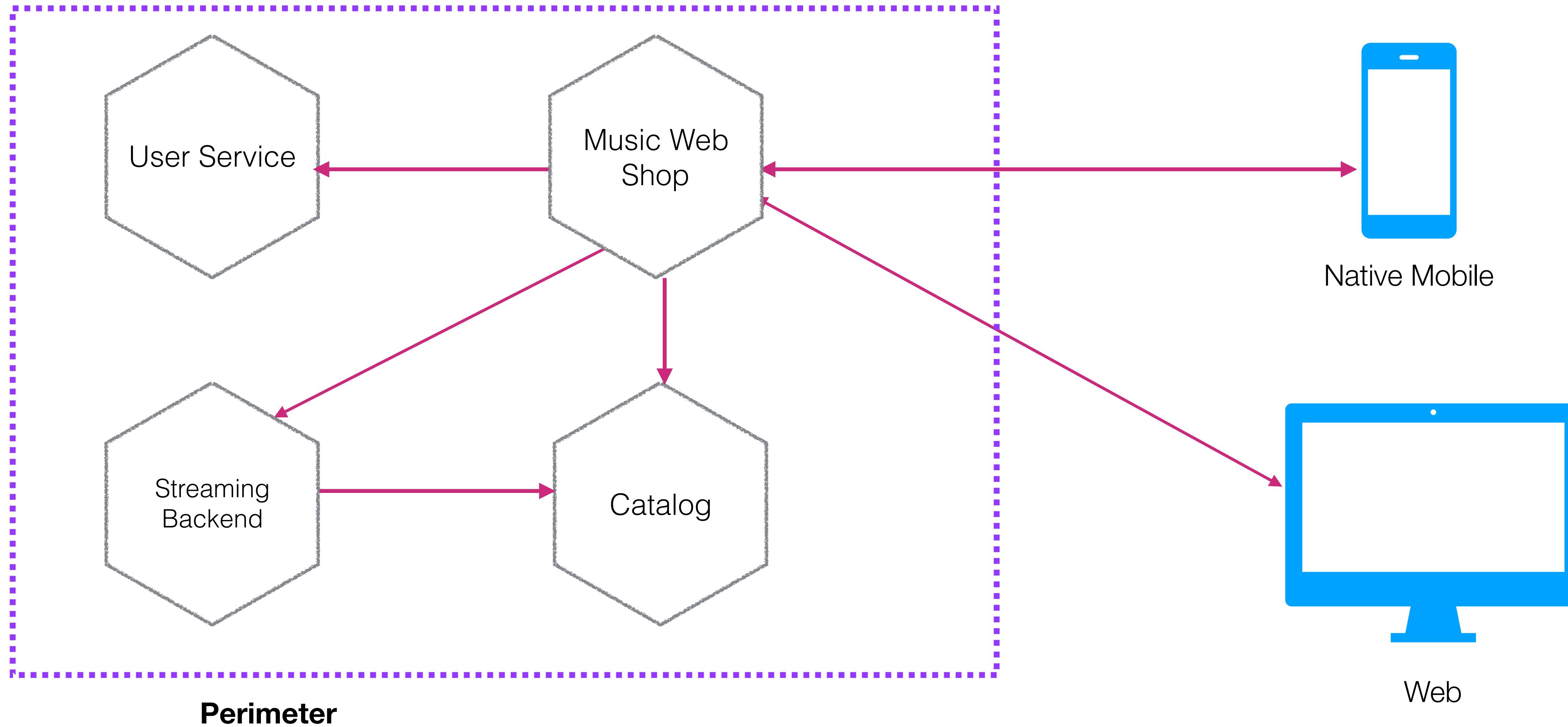
Client-side identity

Server-side identity

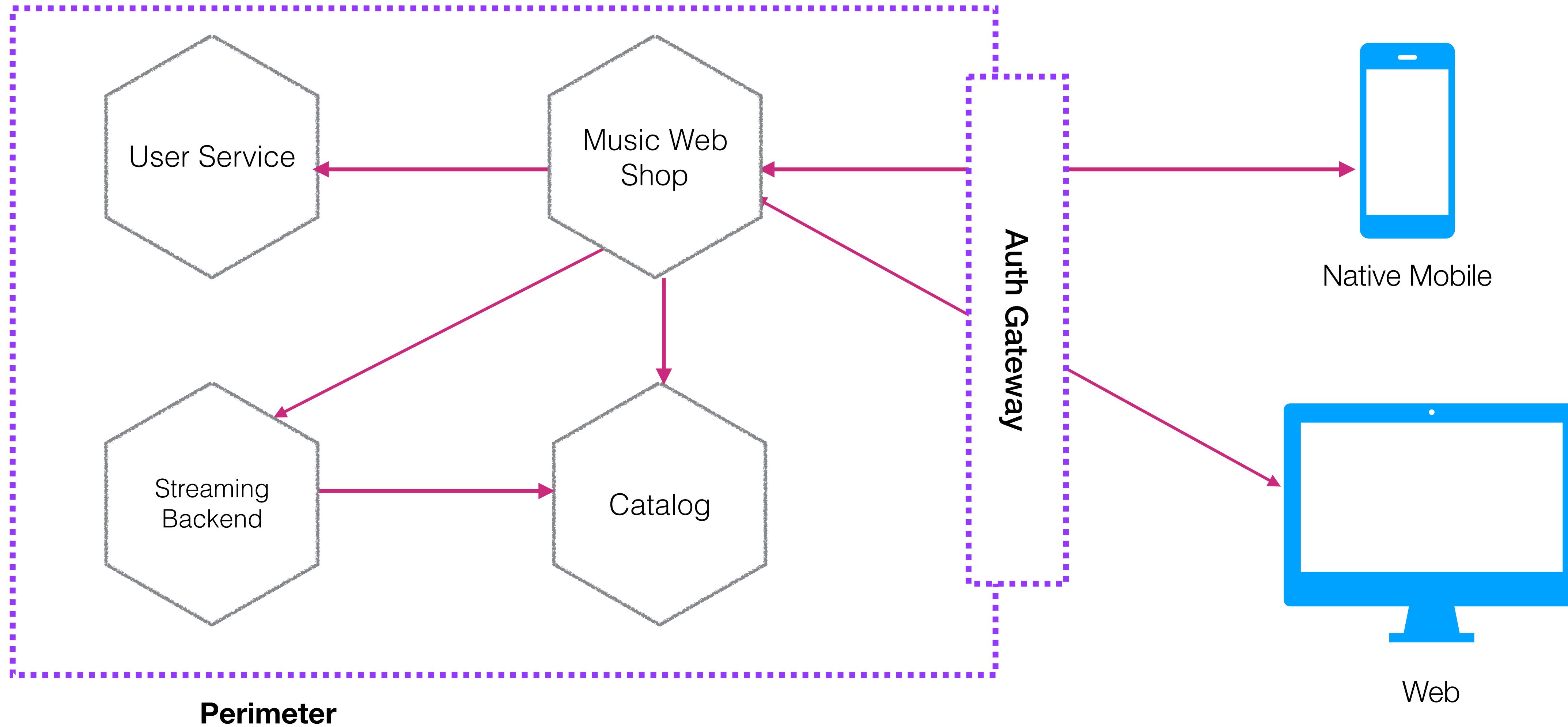
= service-to-service authentication



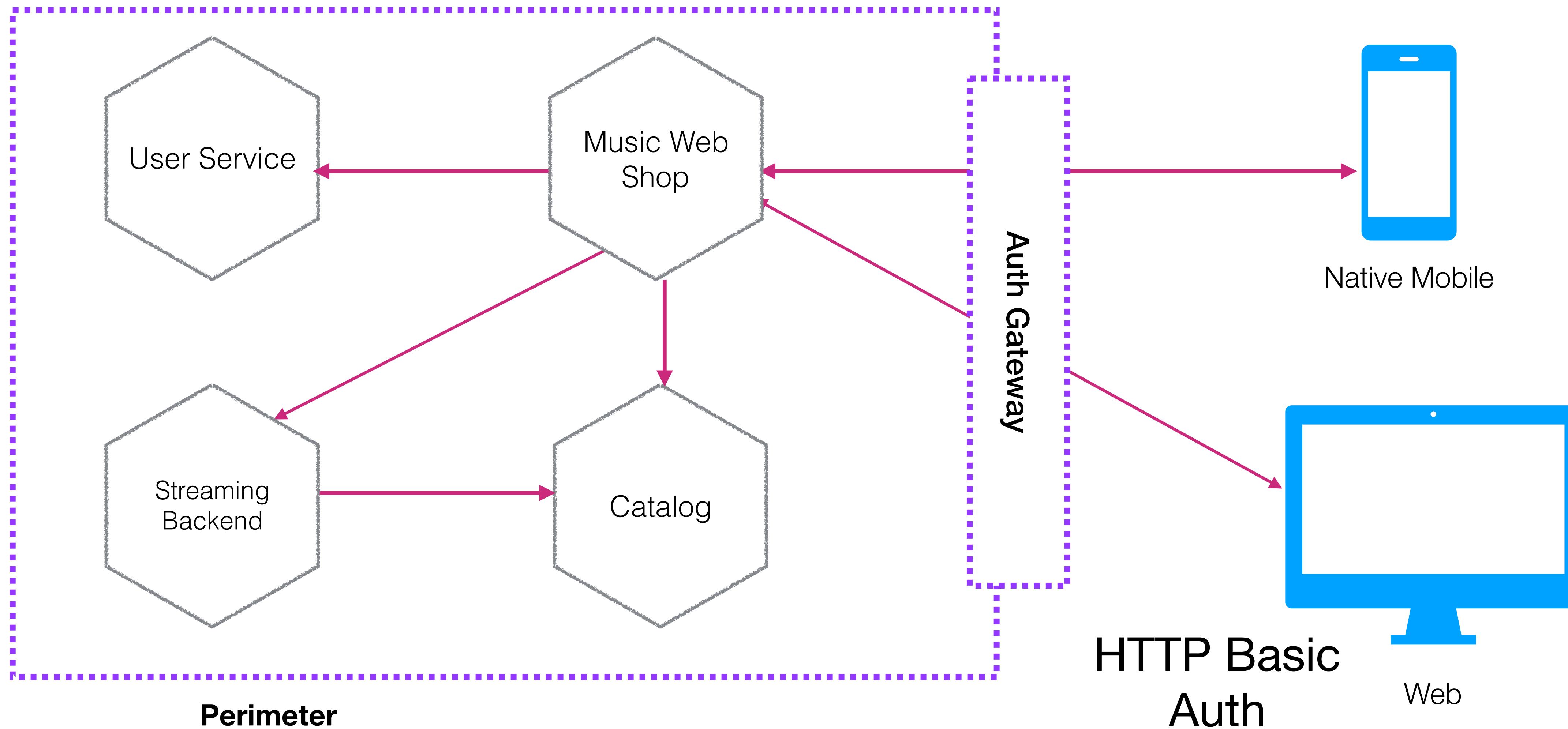
## USER AUTHENTICATION - PROXY-BASED



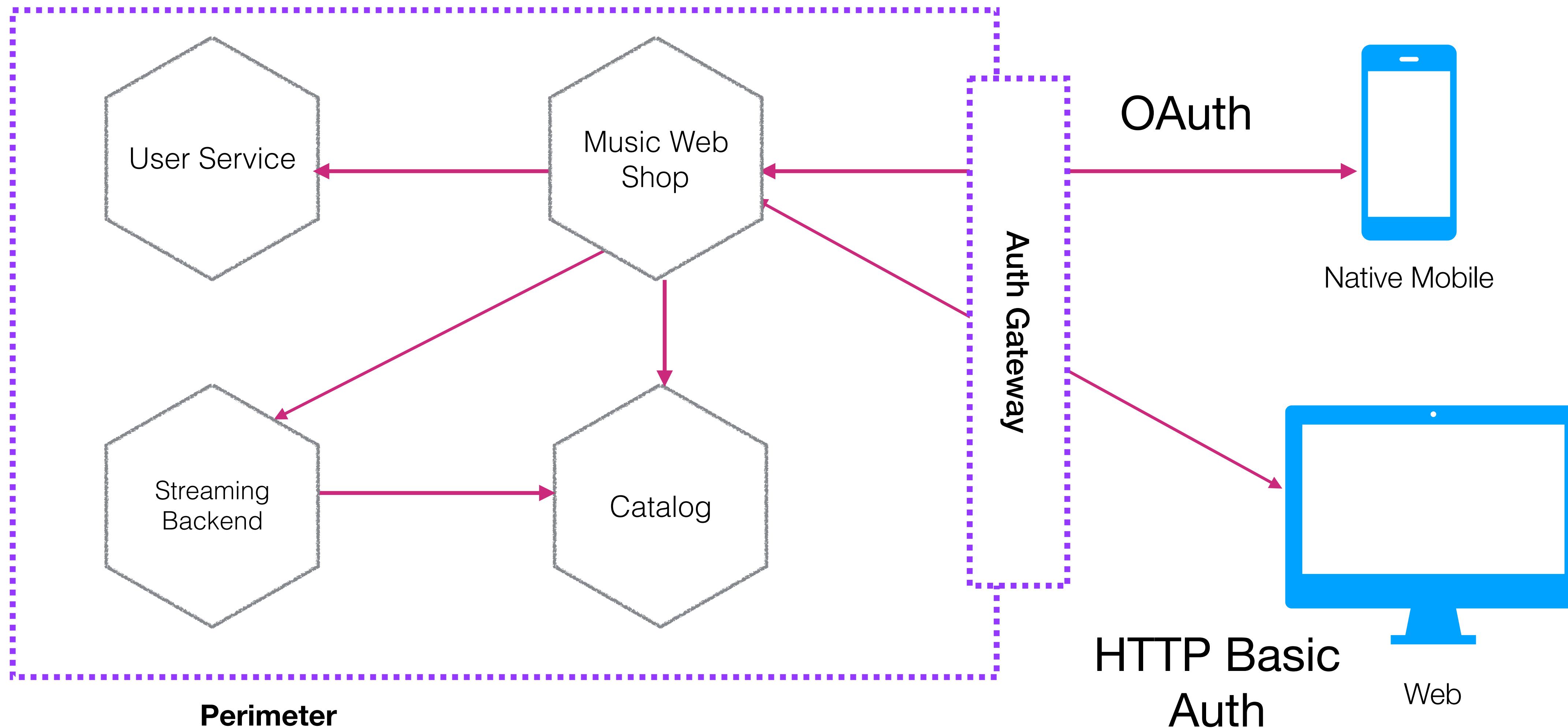
# USER AUTHENTICATION - PROXY-BASED



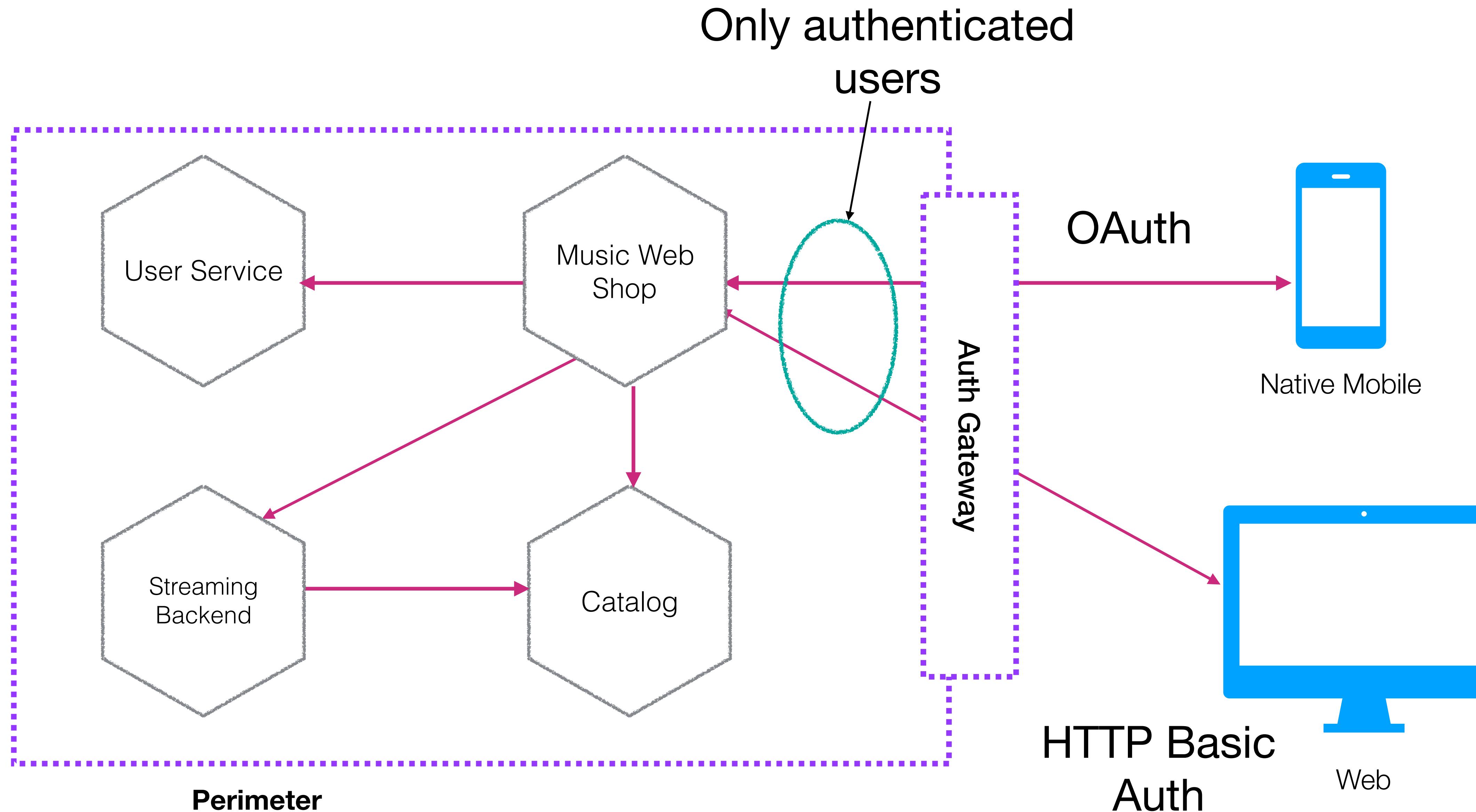
## USER AUTHENTICATION - PROXY-BASED



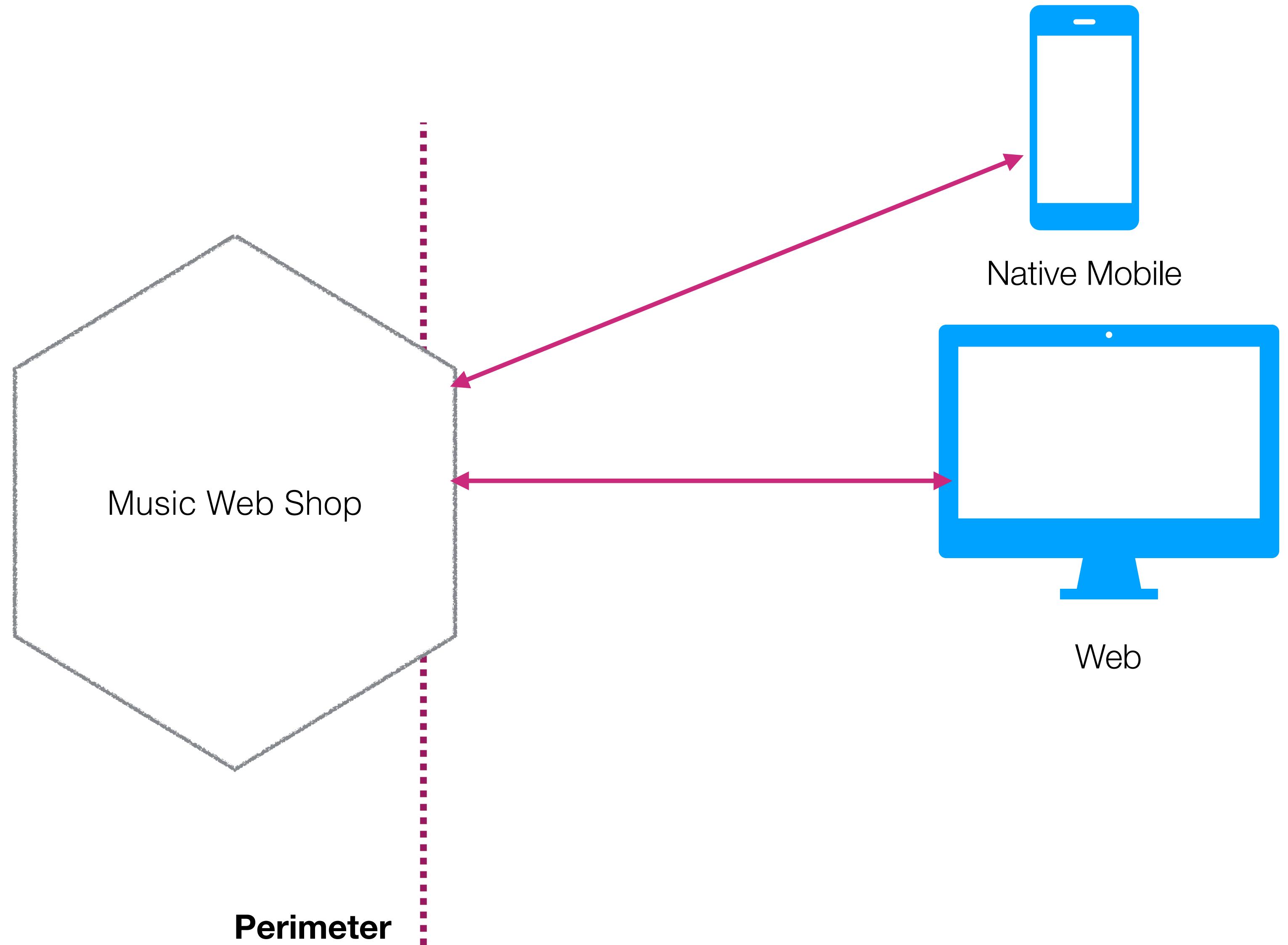
## USER AUTHENTICATION - PROXY-BASED



## USER AUTHENTICATION - PROXY-BASED

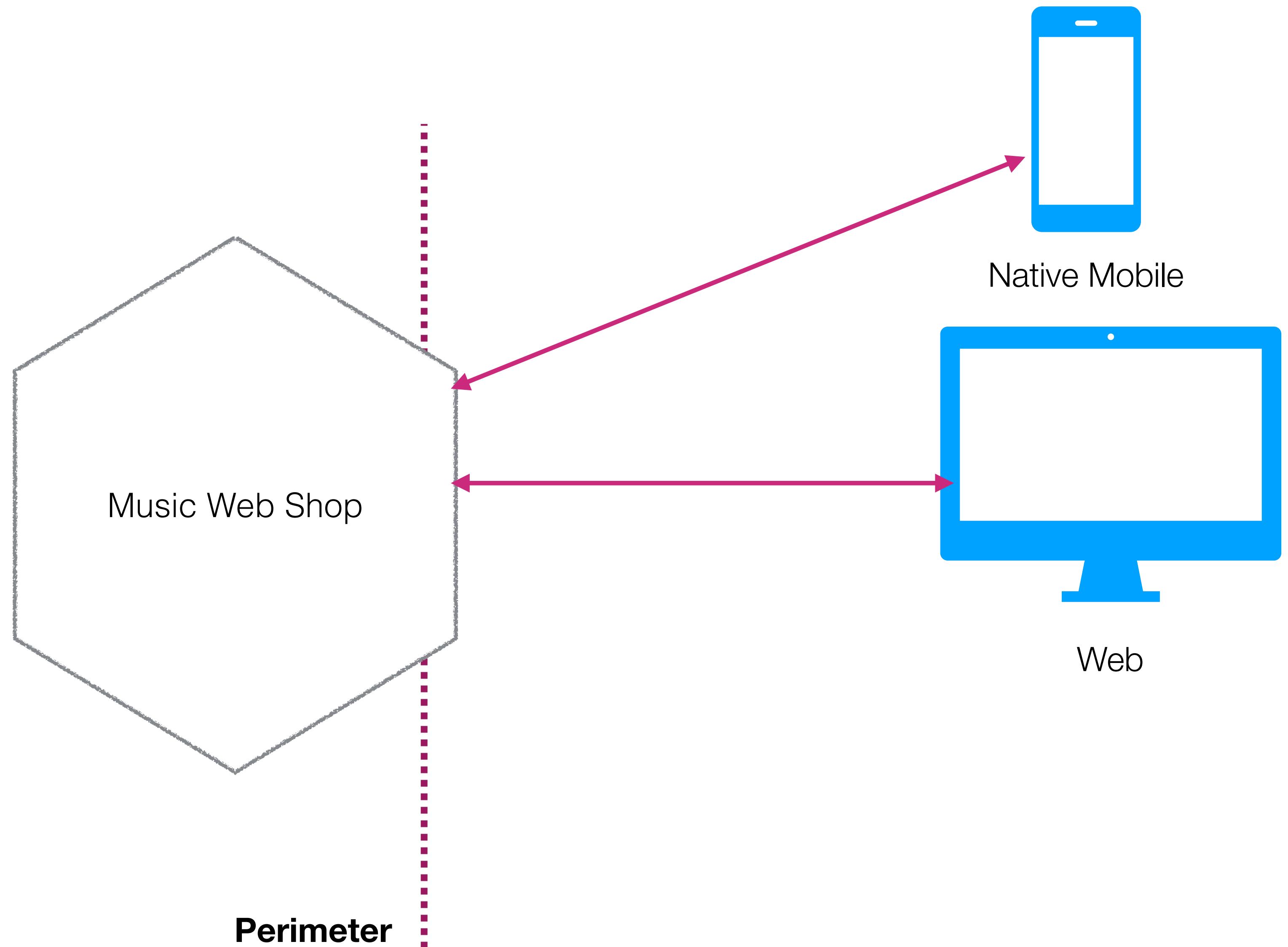


## USER AUTHENTICATION - HANDLED IN SERVICE



## USER AUTHENTICATION - HANDLED IN SERVICE

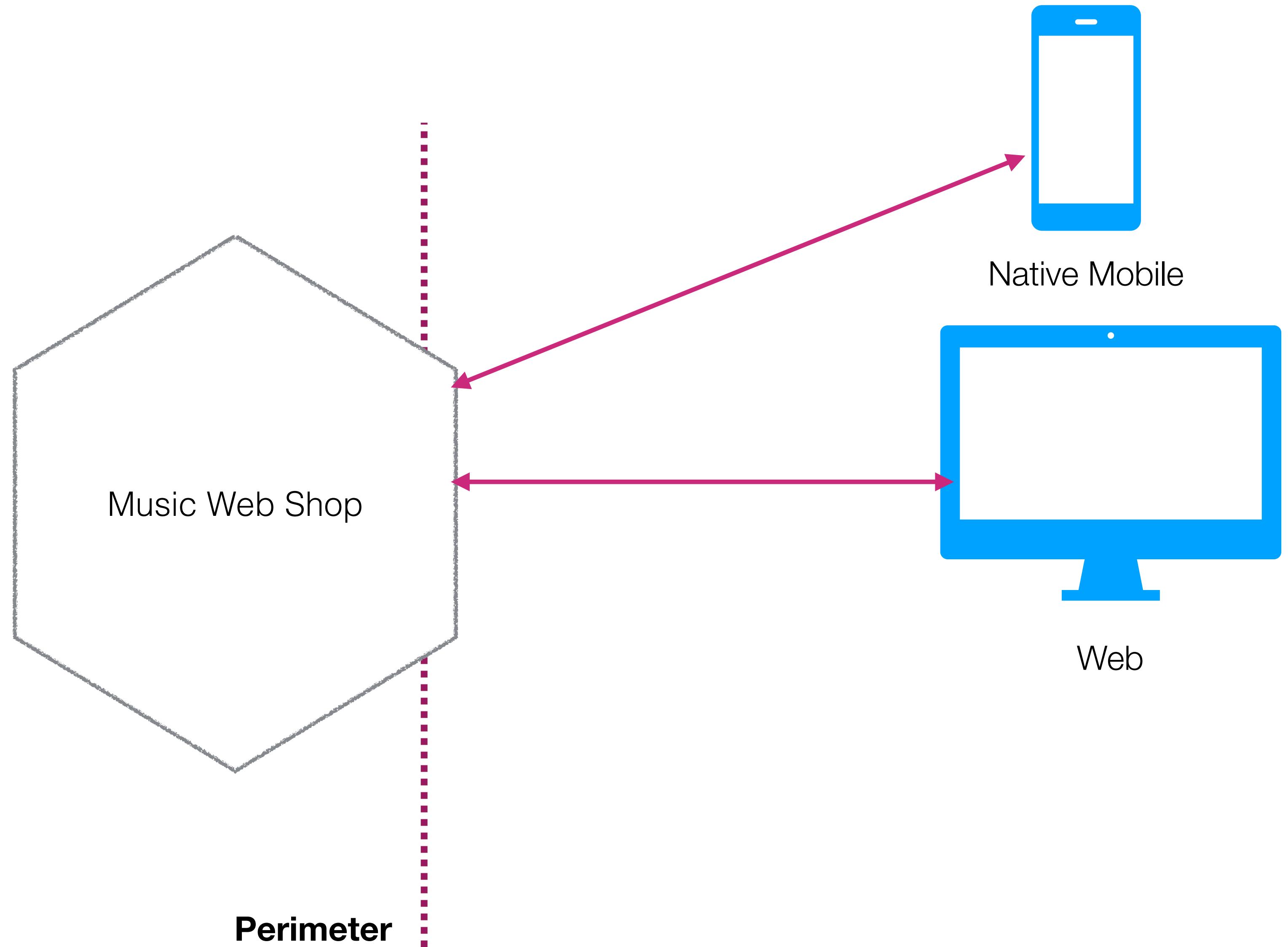
Can reduce latency



## USER AUTHENTICATION - HANDLED IN SERVICE

Can reduce latency

Service potentially exposed to public internet

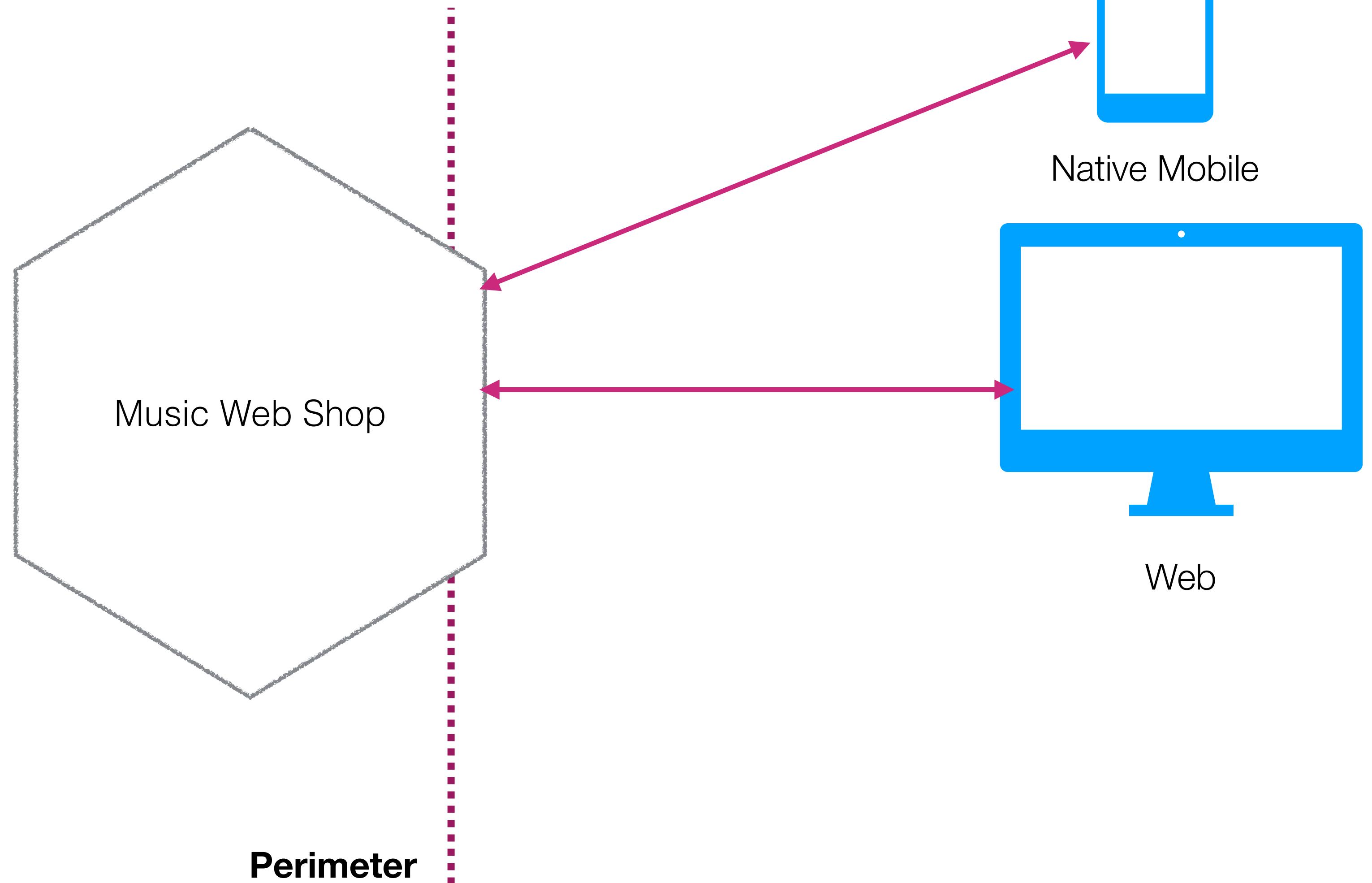


## USER AUTHENTICATION - HANDLED IN SERVICE

Can reduce latency

Service potentially exposed to public internet

Self-contained



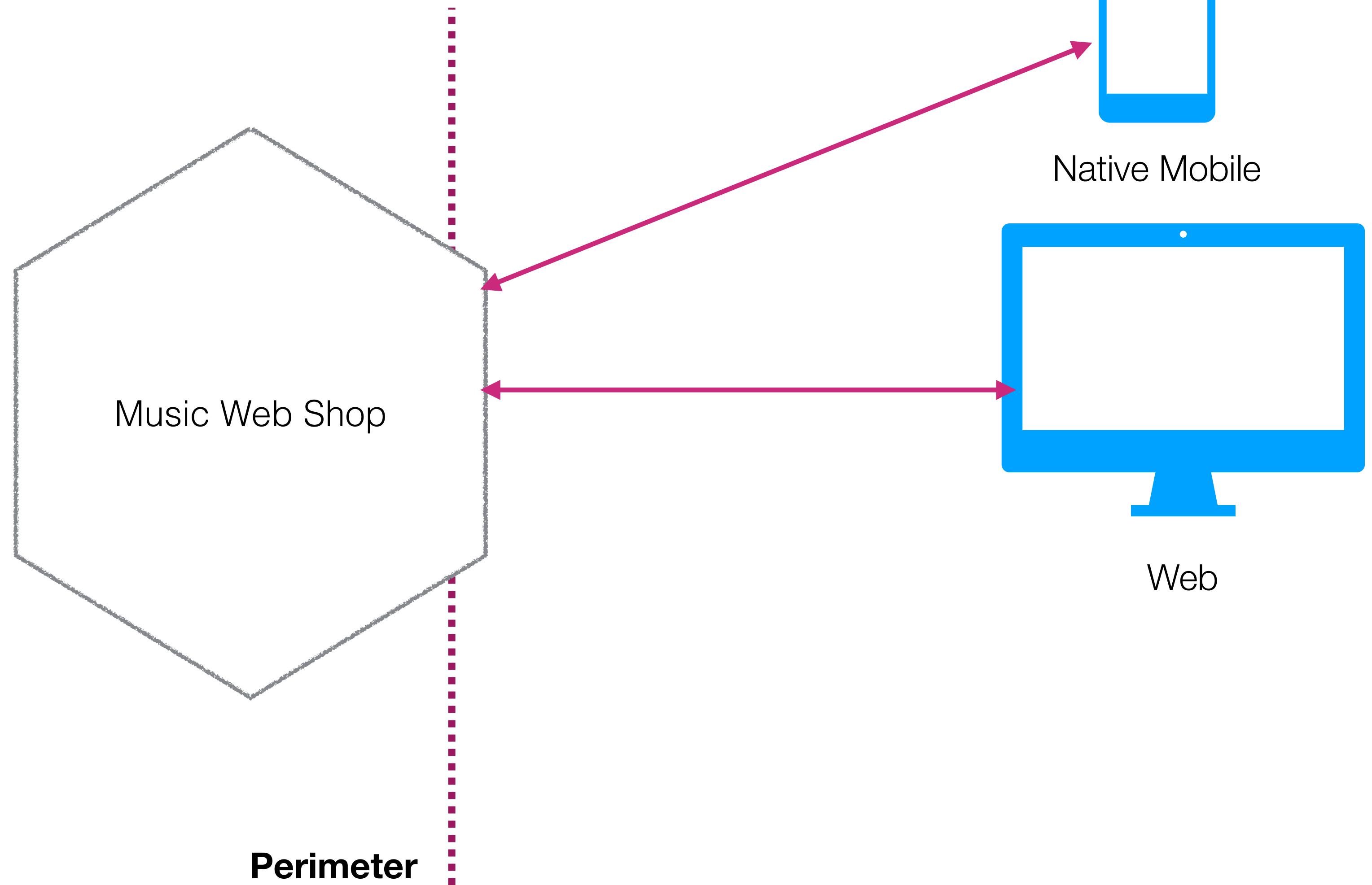
## USER AUTHENTICATION - HANDLED IN SERVICE

Can reduce latency

Service potentially exposed to public internet

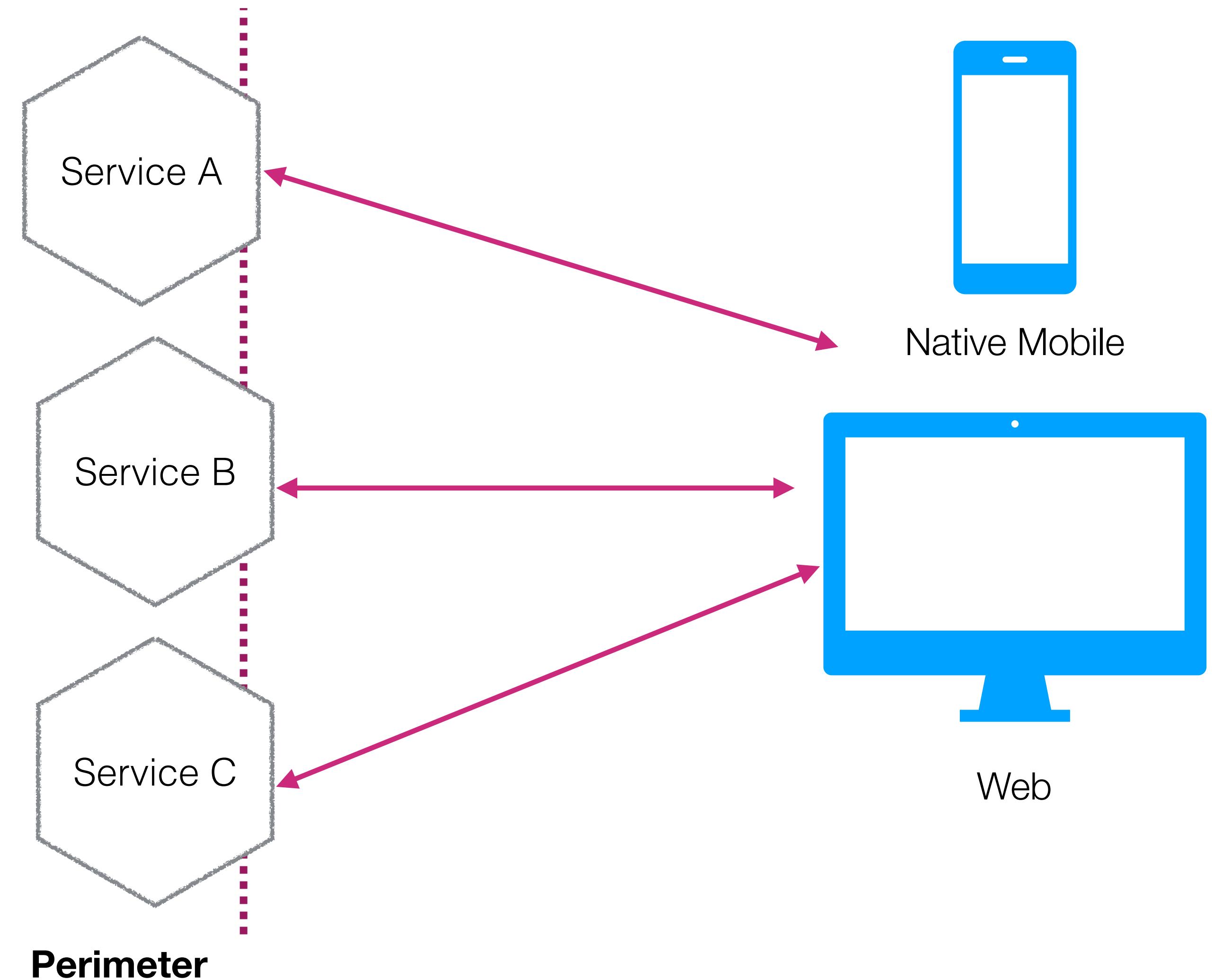
Self-contained

Code reuse?



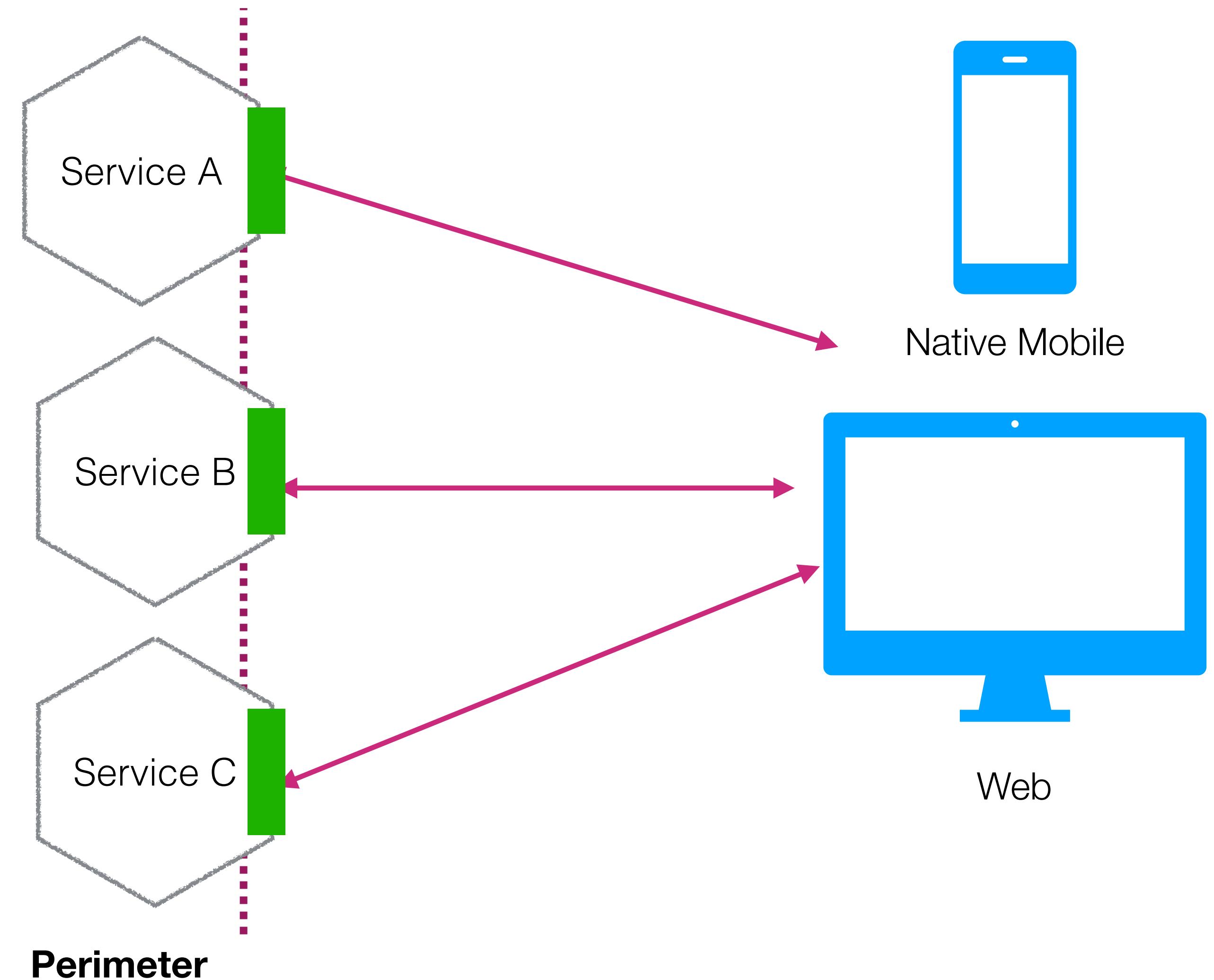
## USER AUTHENTICATION - LIBRARY-BASED

Re-use authentication  
flow code via library



## USER AUTHENTICATION - LIBRARY-BASED

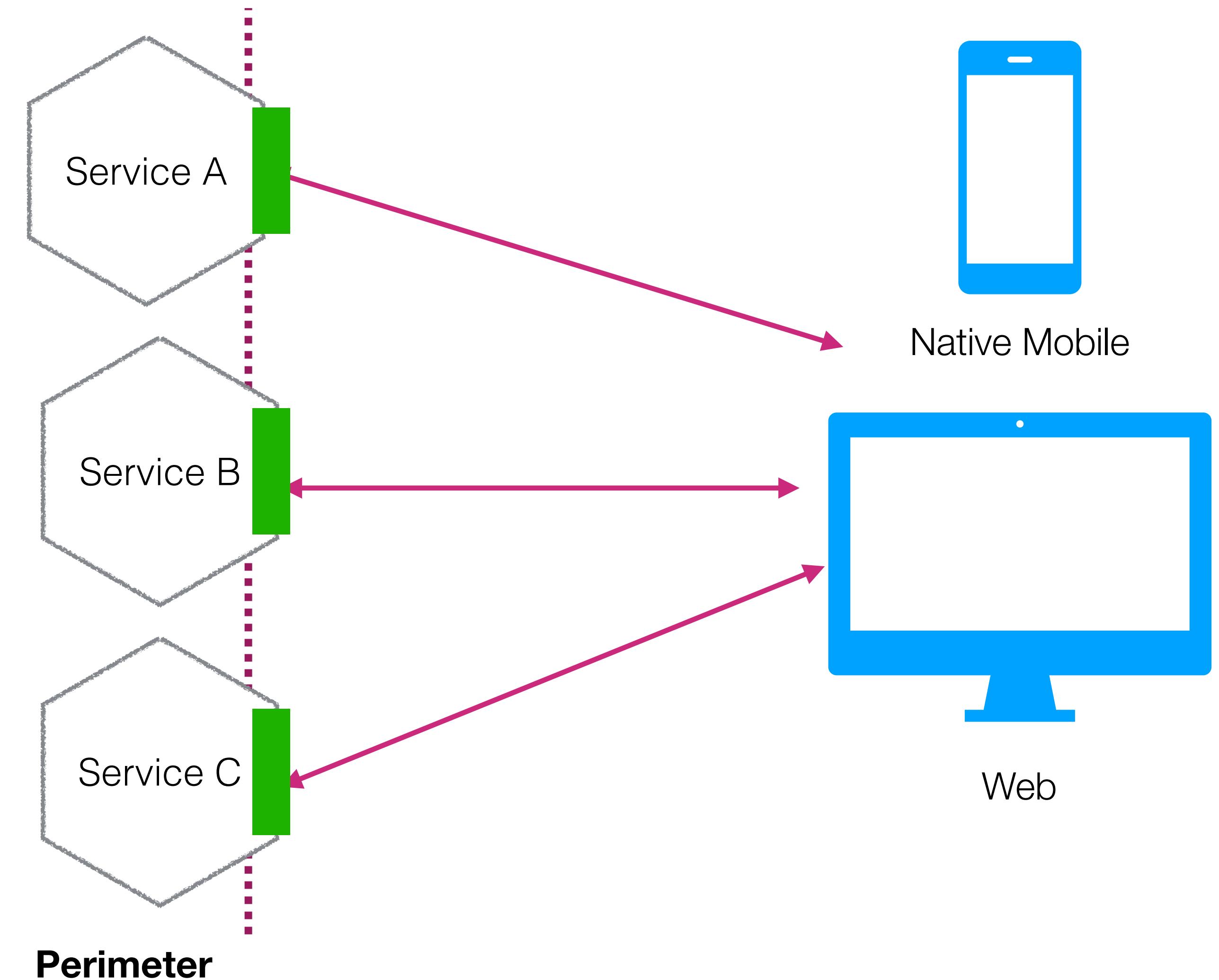
Re-use authentication  
flow code via library



## USER AUTHENTICATION - LIBRARY-BASED

Re-use authentication  
flow code via library

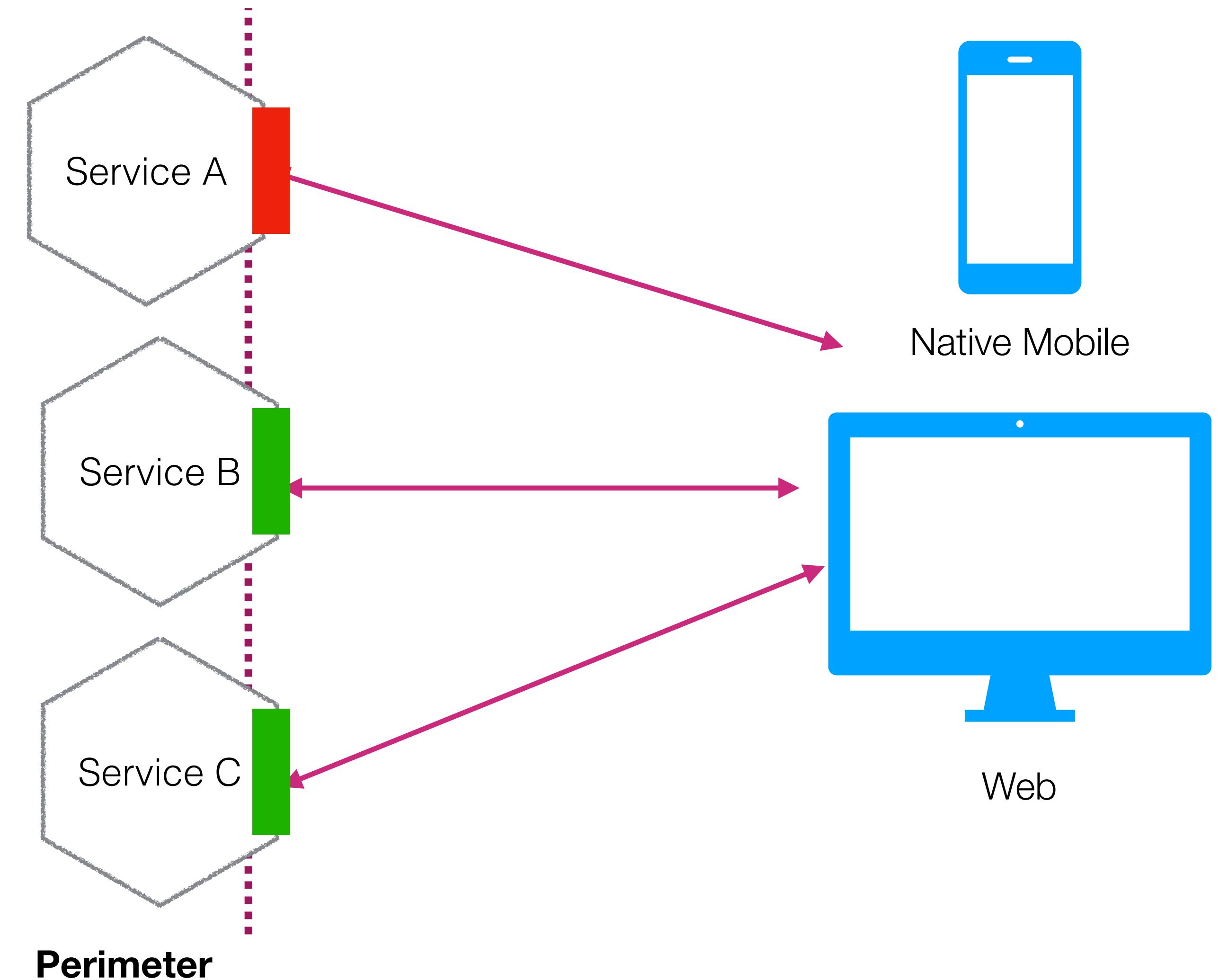
Version drift?



## USER AUTHENTICATION - LIBRARY-BASED

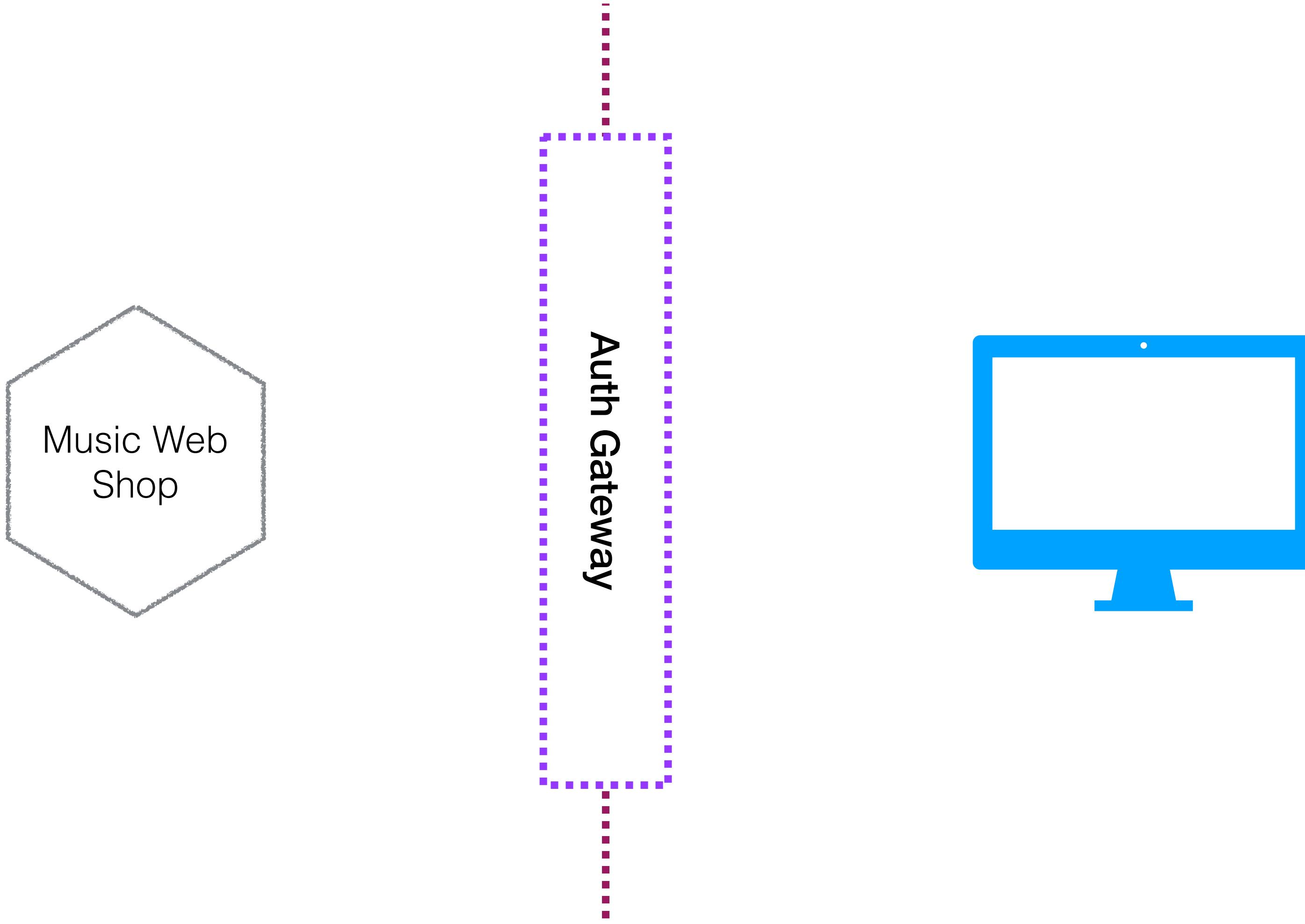
Re-use authentication  
flow code via library

Version drift?

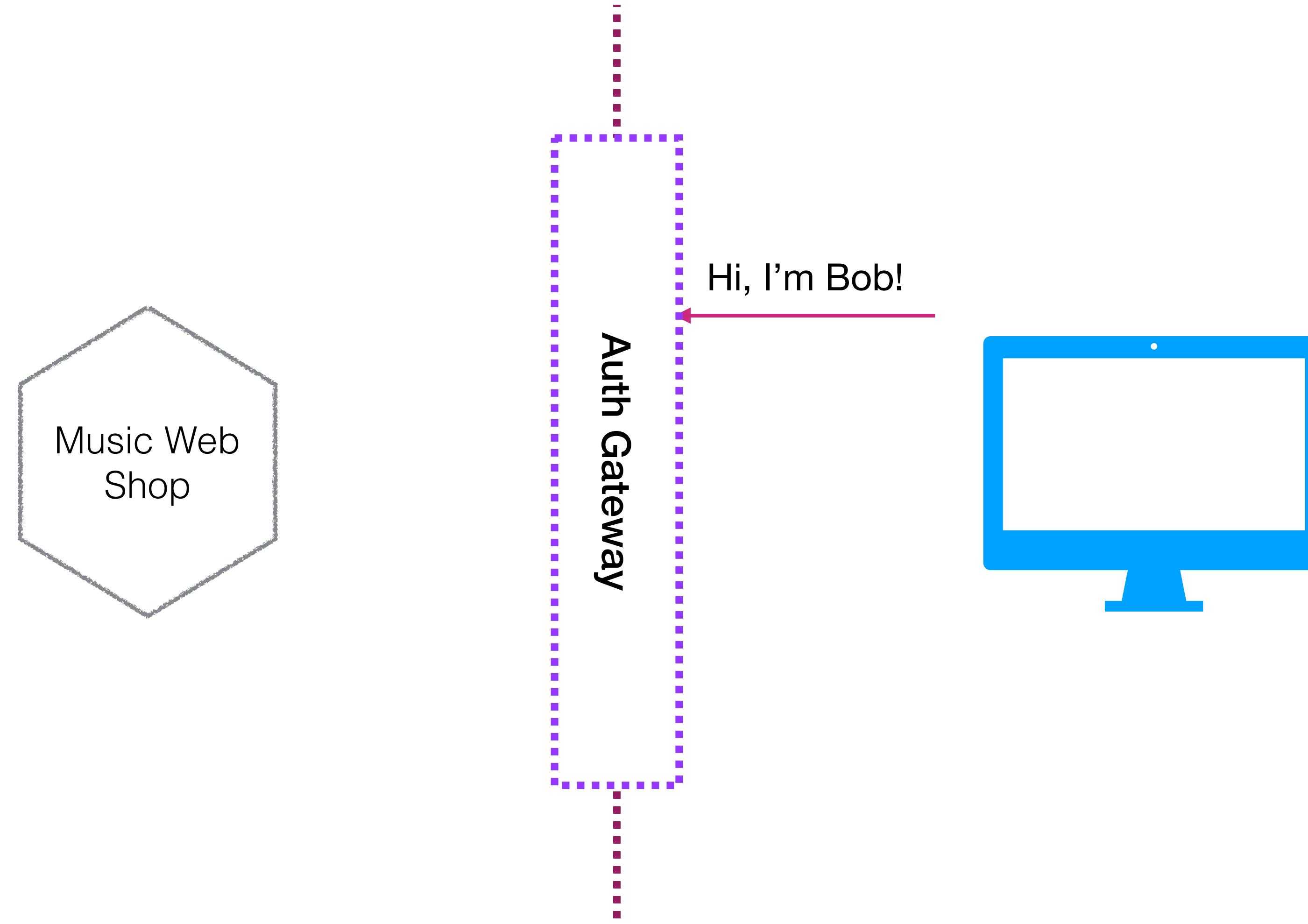


# What about authorisation?

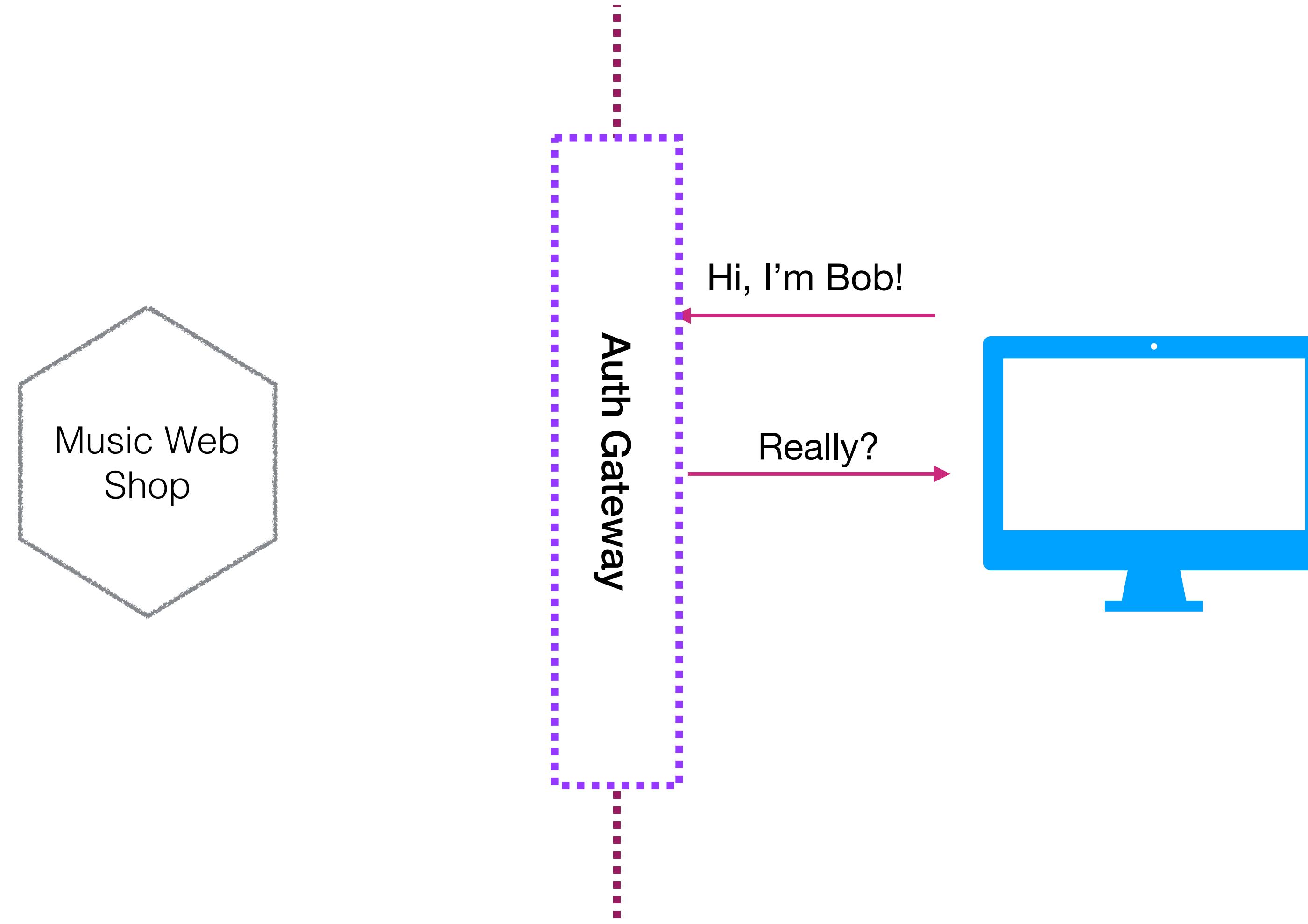
# DO YOU EVEN AUTH?



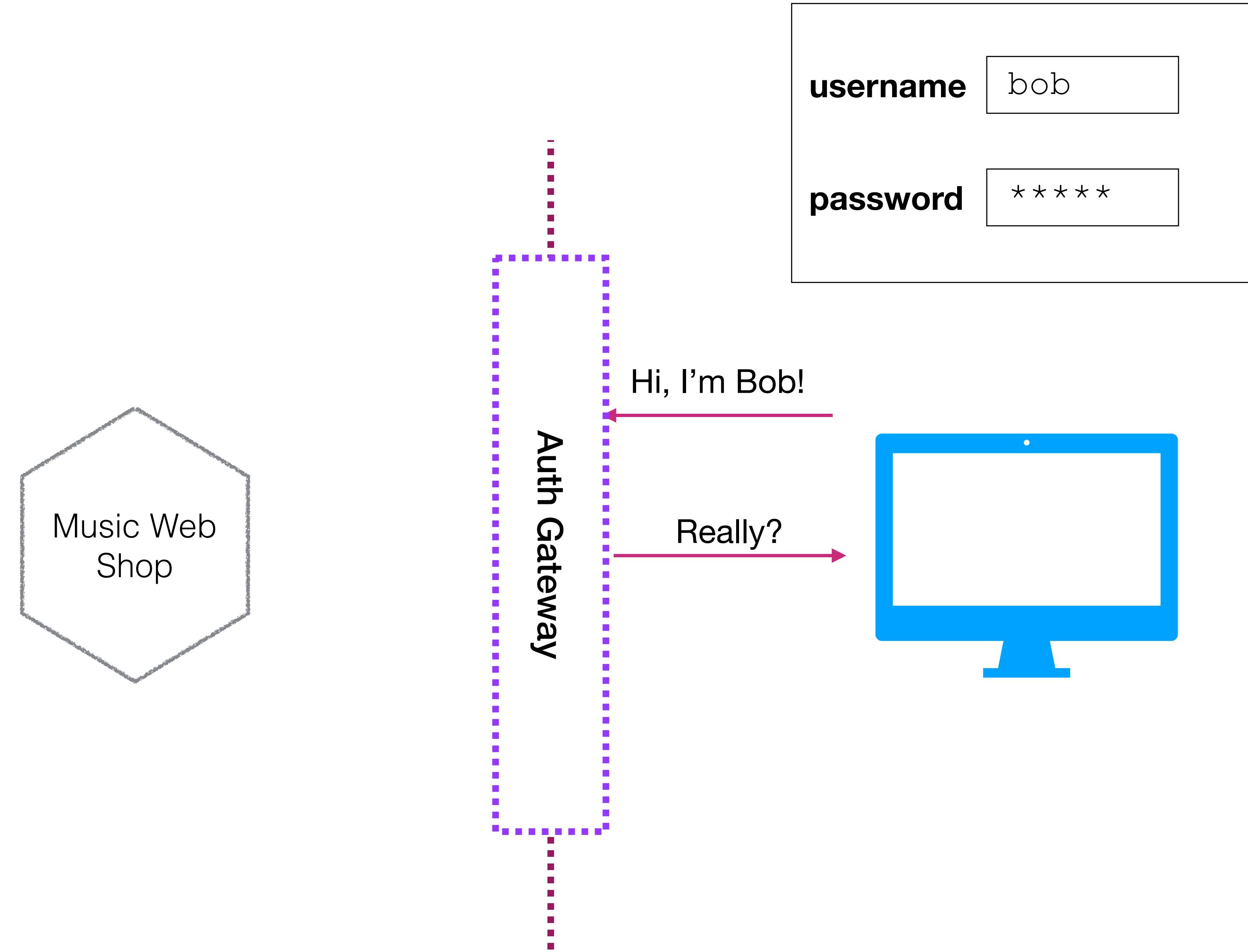
# DO YOU EVEN AUTH?



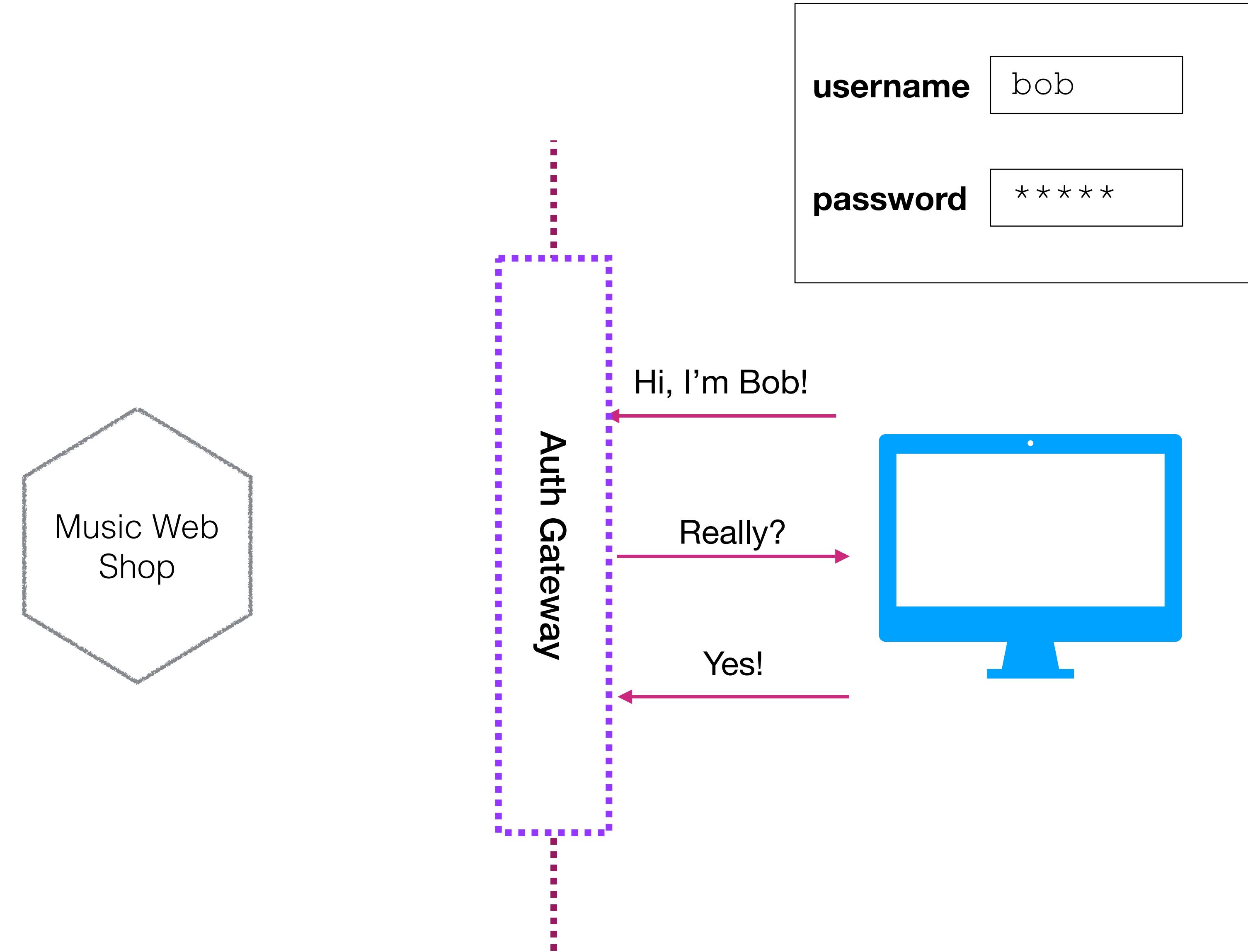
# DO YOU EVEN AUTH?



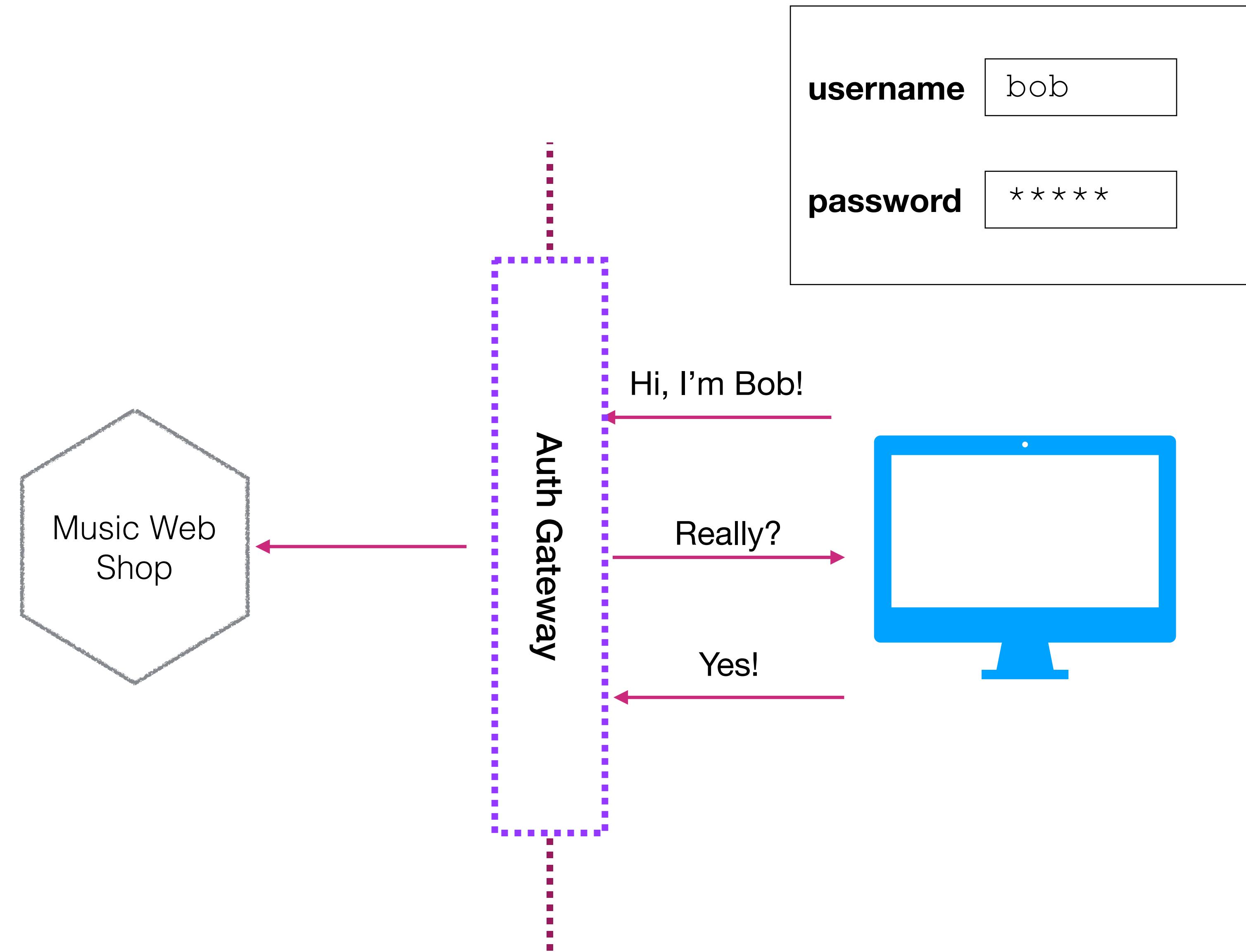
# DO YOU EVEN AUTH?



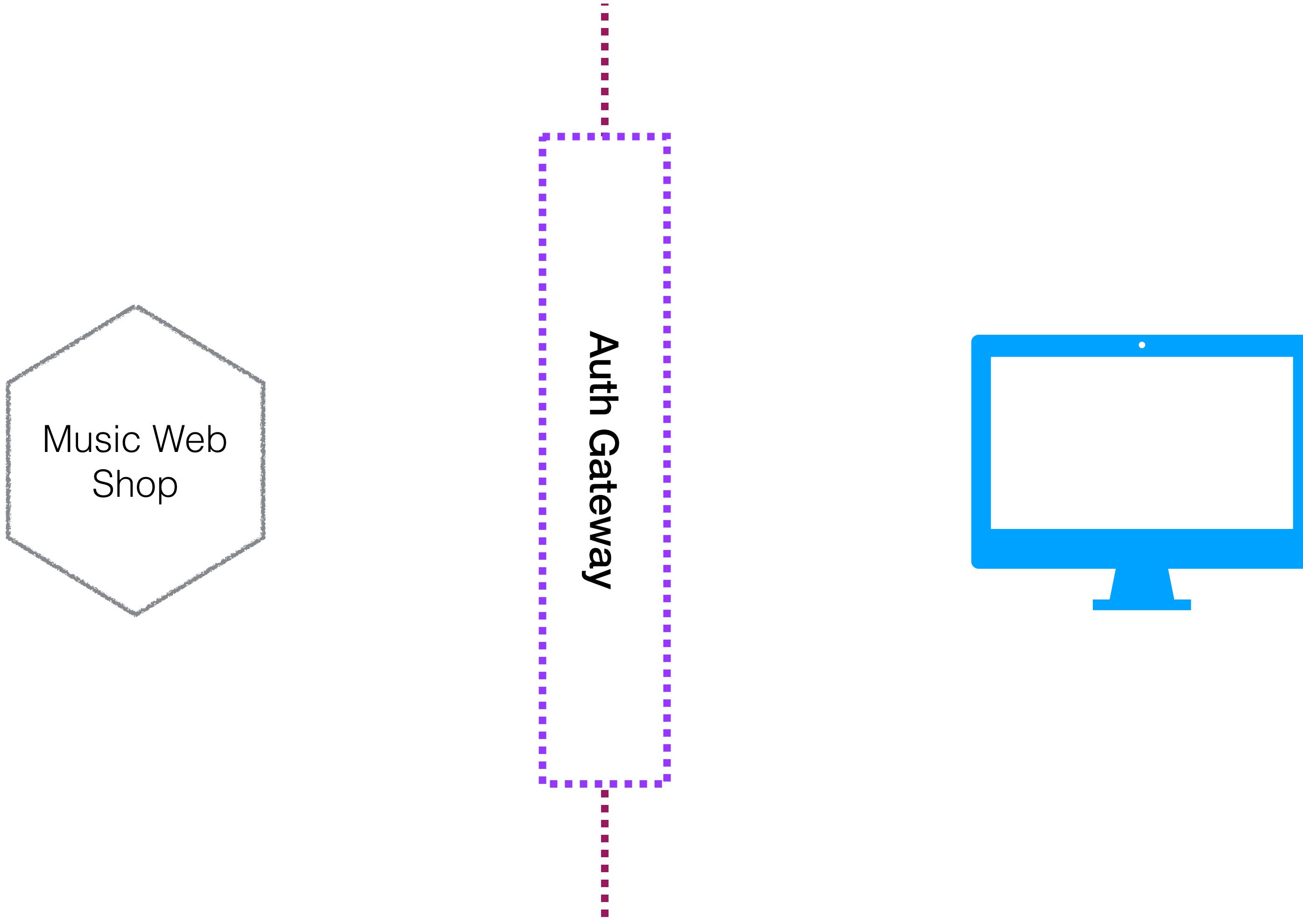
# DO YOU EVEN AUTH?



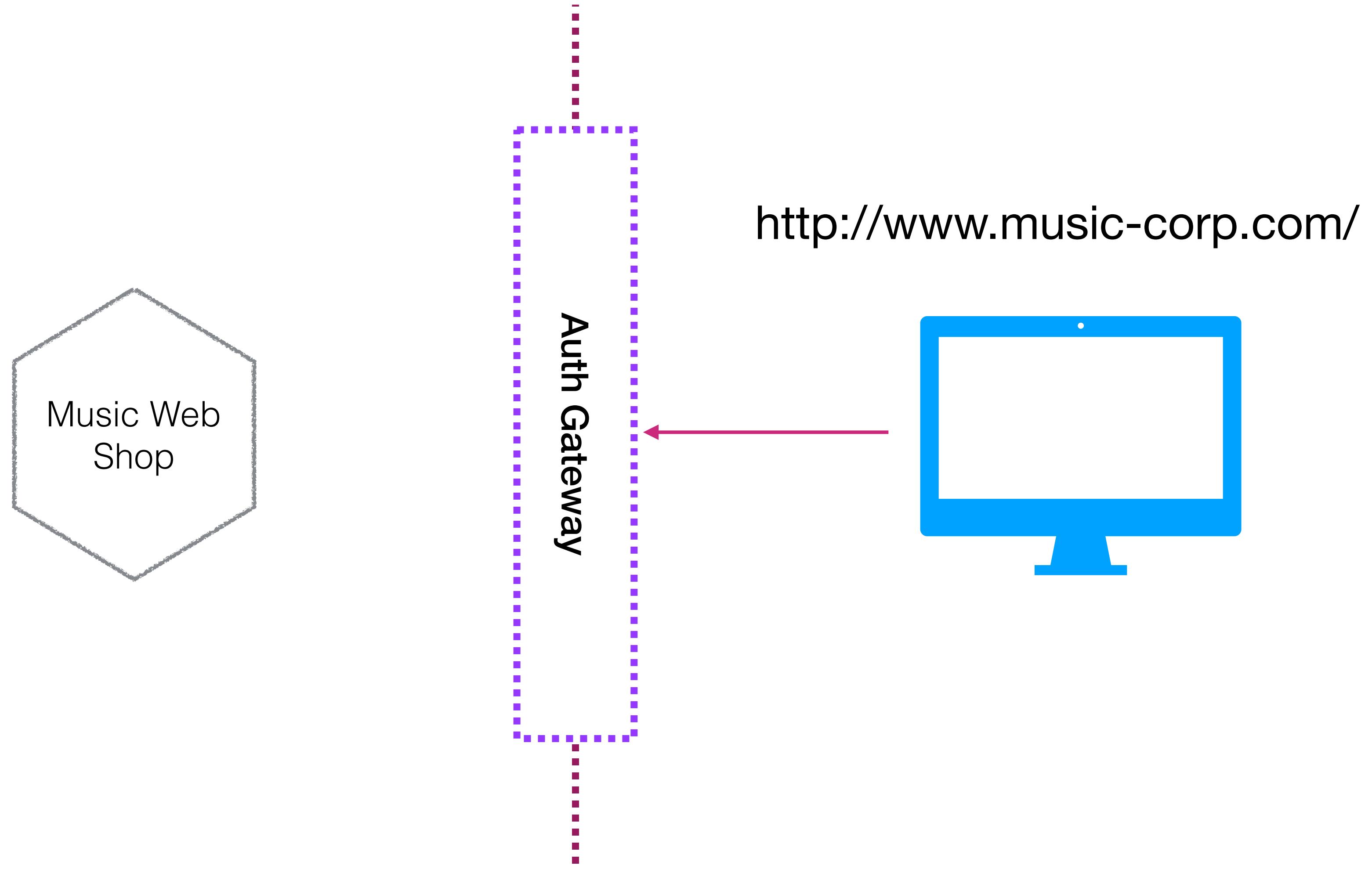
# DO YOU EVEN AUTH?



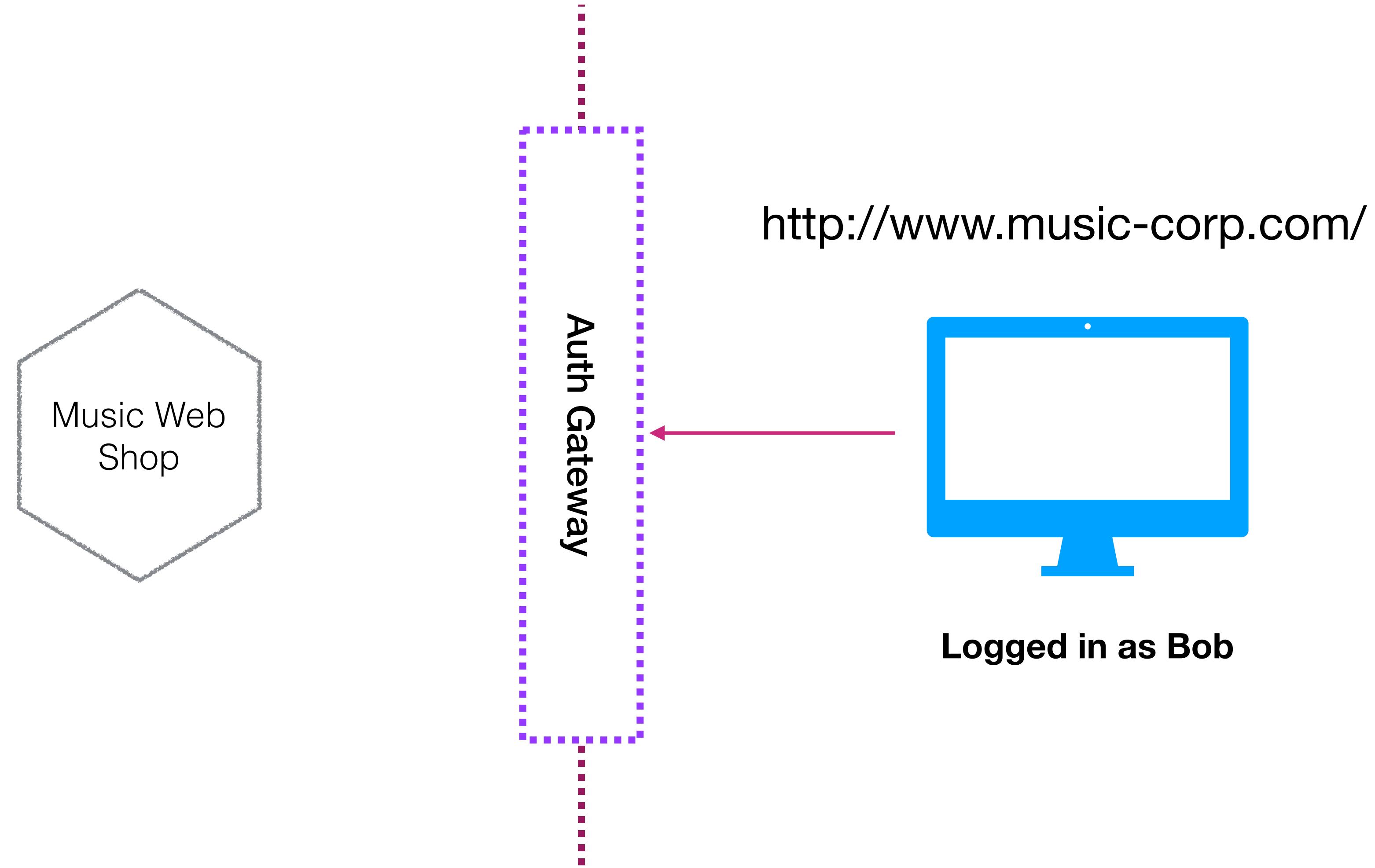
# DO YOU EVEN AUTH?



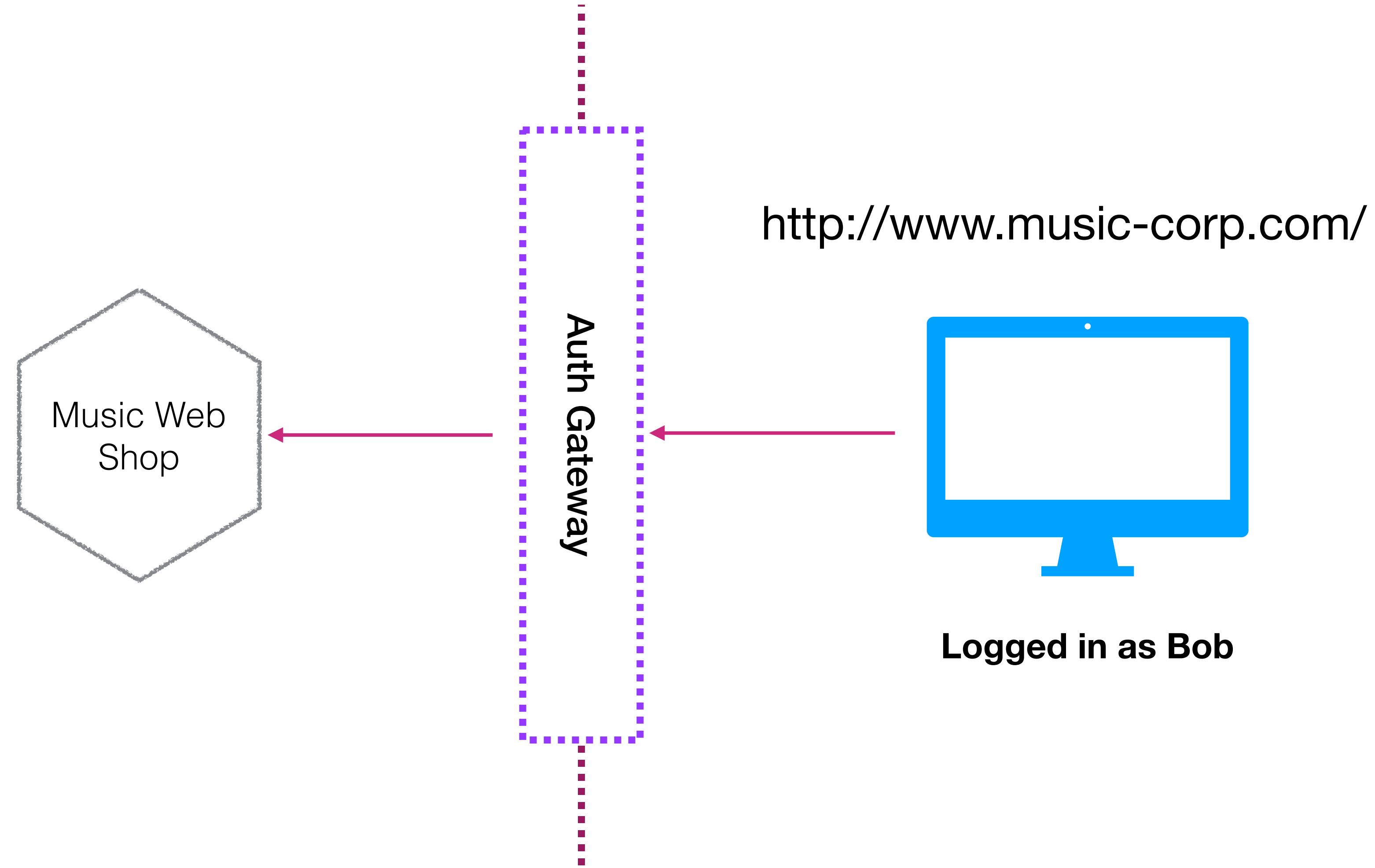
# DO YOU EVEN AUTH?



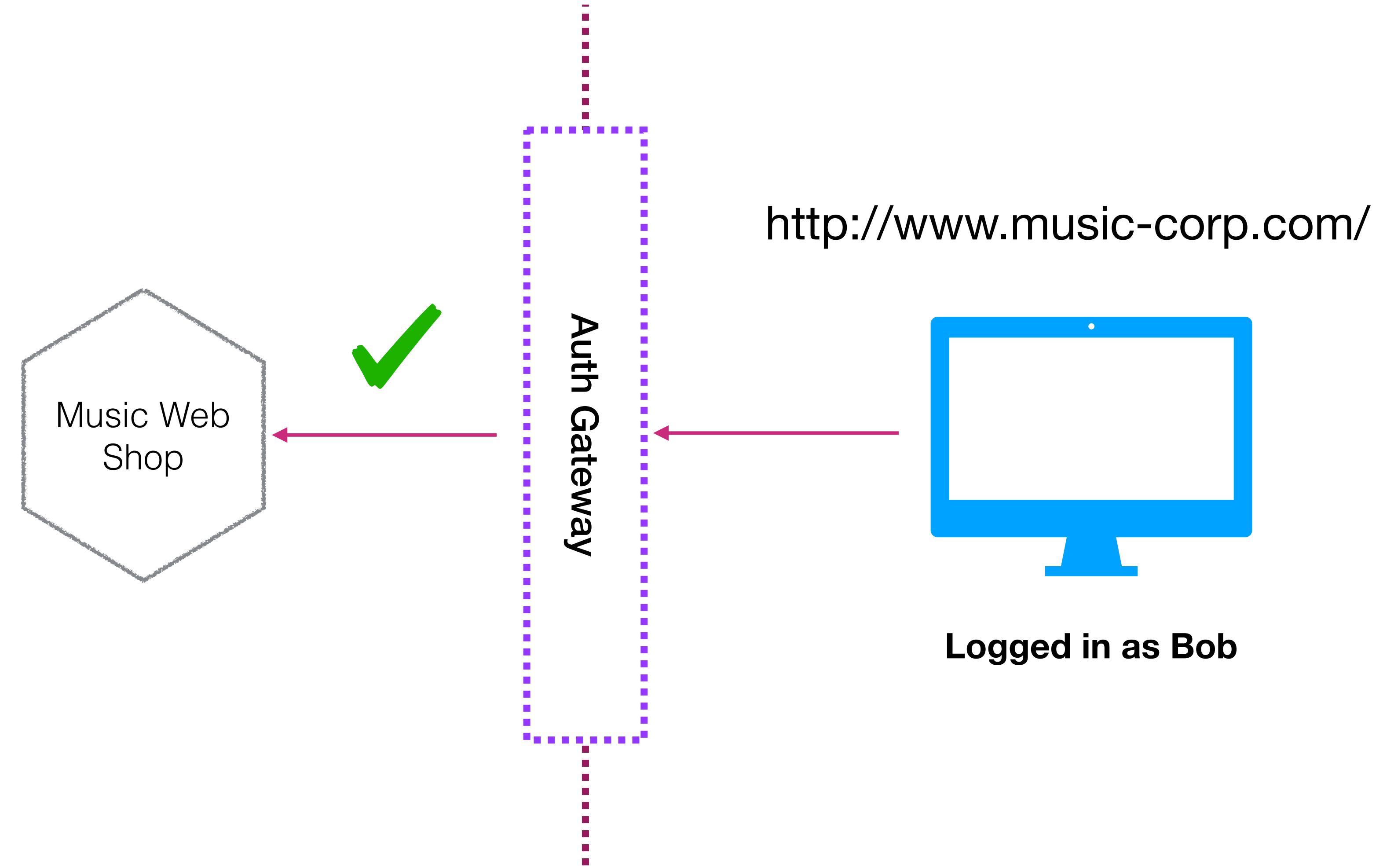
# DO YOU EVEN AUTH?



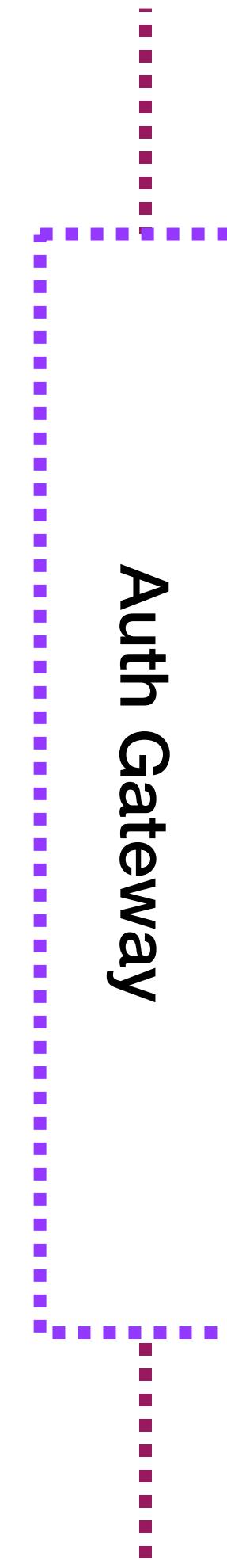
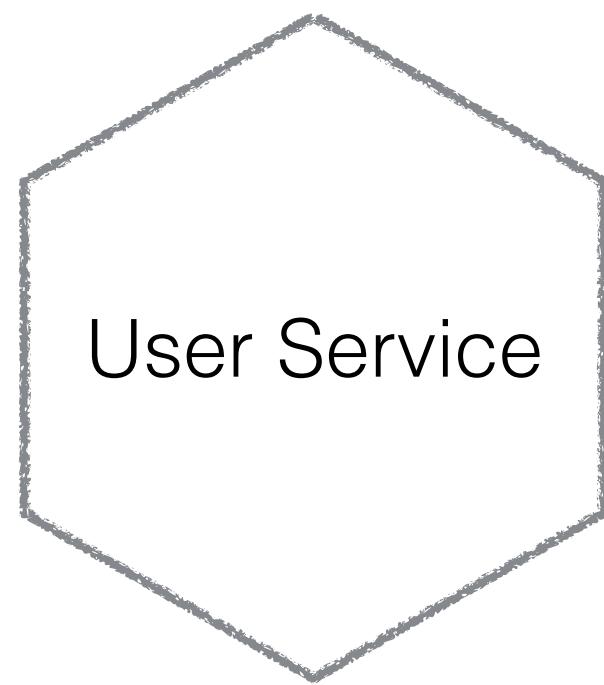
# DO YOU EVEN AUTH?



# DO YOU EVEN AUTH?



# DOWNSTREAM AUTH - IMPLICIT TRUST?

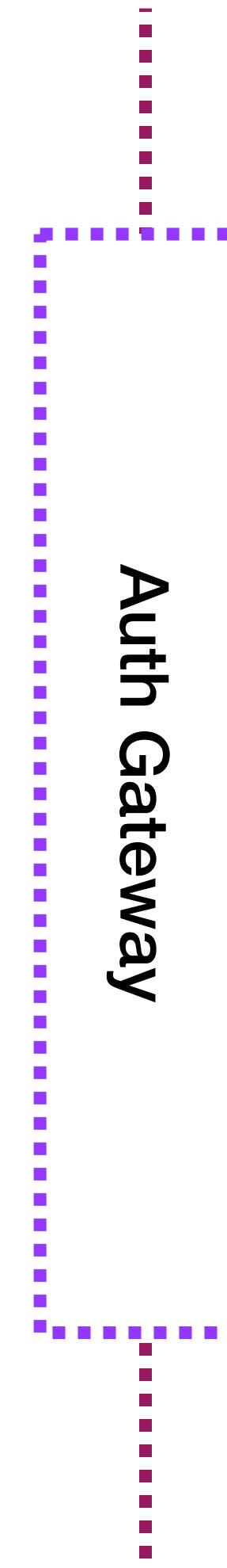
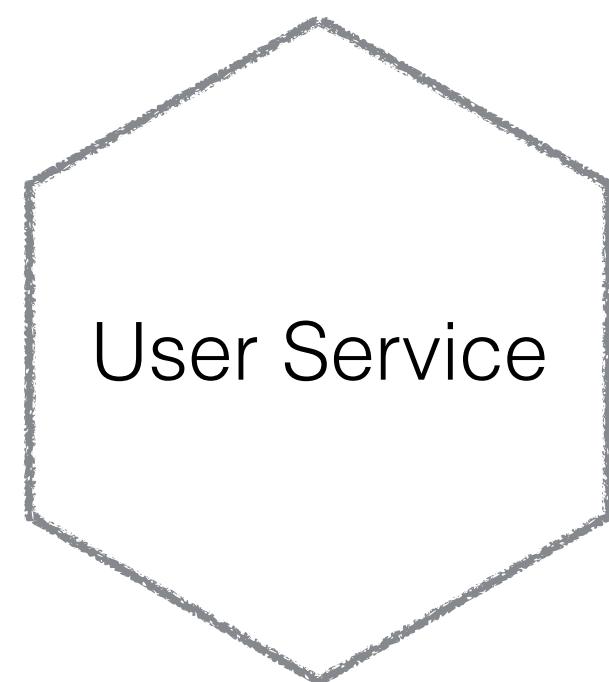


Auth Gateway



**Logged in as Bob**

## DOWNSTREAM AUTH - IMPLICIT TRUST?

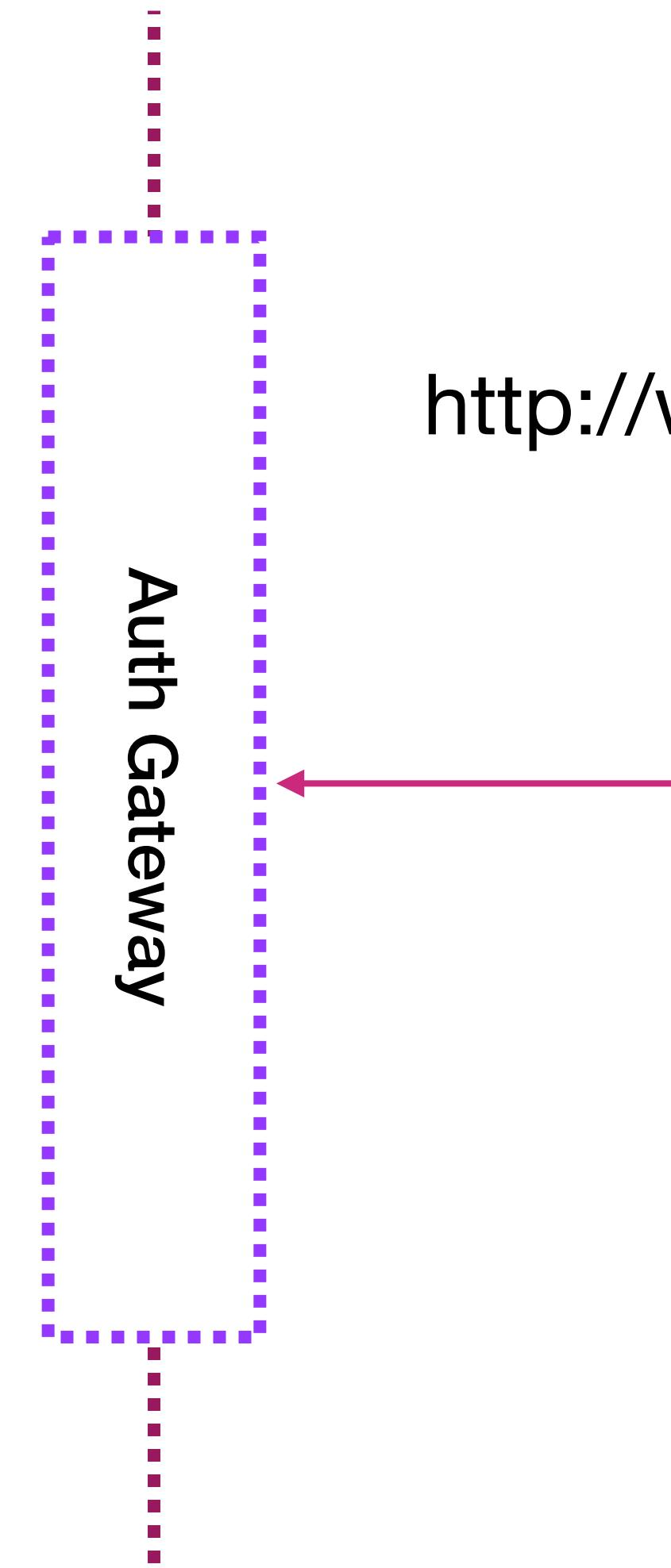
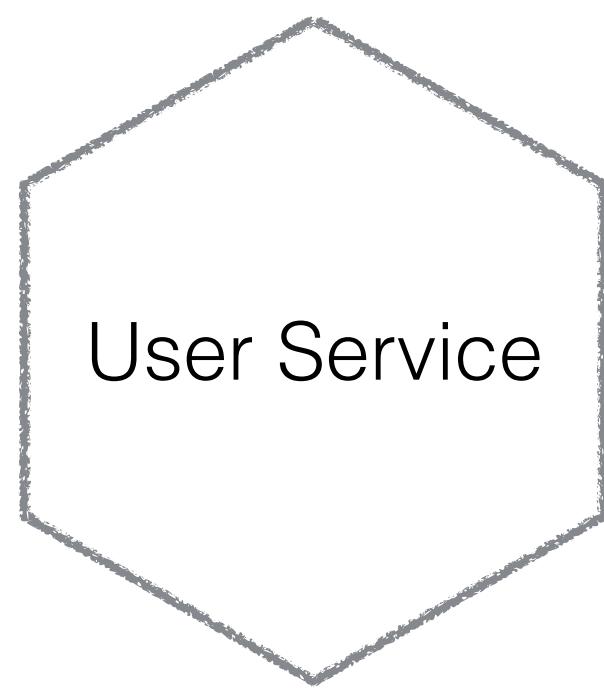


`http://www.music-corp.com/user/bob`



**Logged in as Bob**

## DOWNSTREAM AUTH - IMPLICIT TRUST?

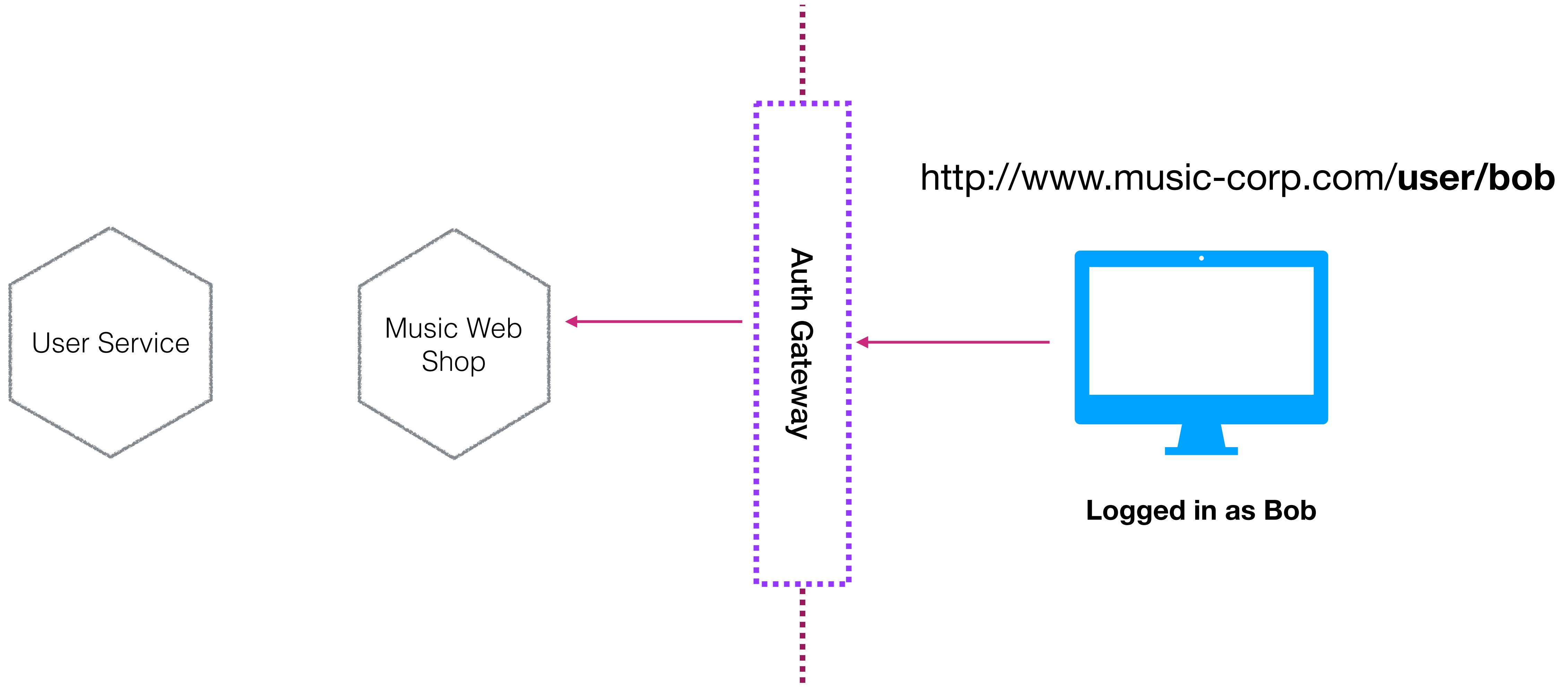


`http://www.music-corp.com/user/bob`

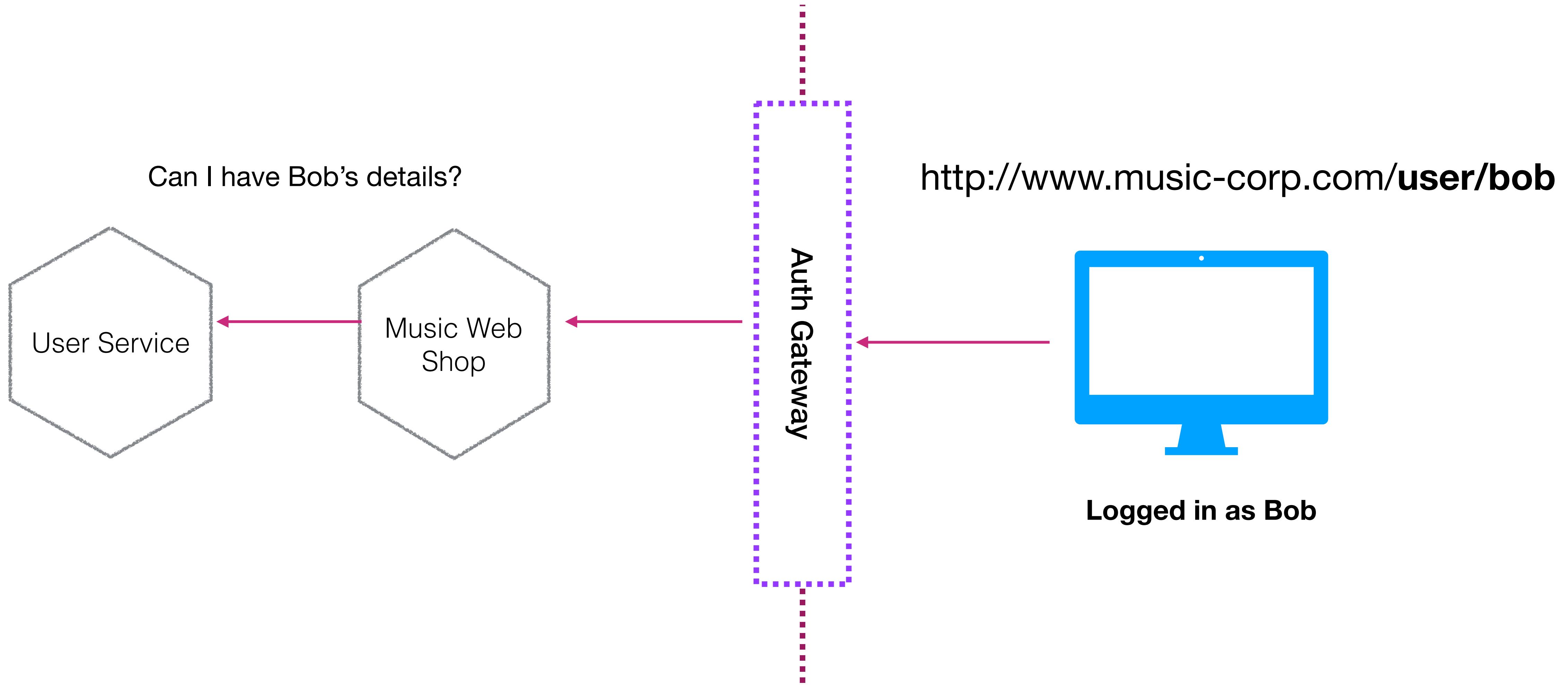


**Logged in as Bob**

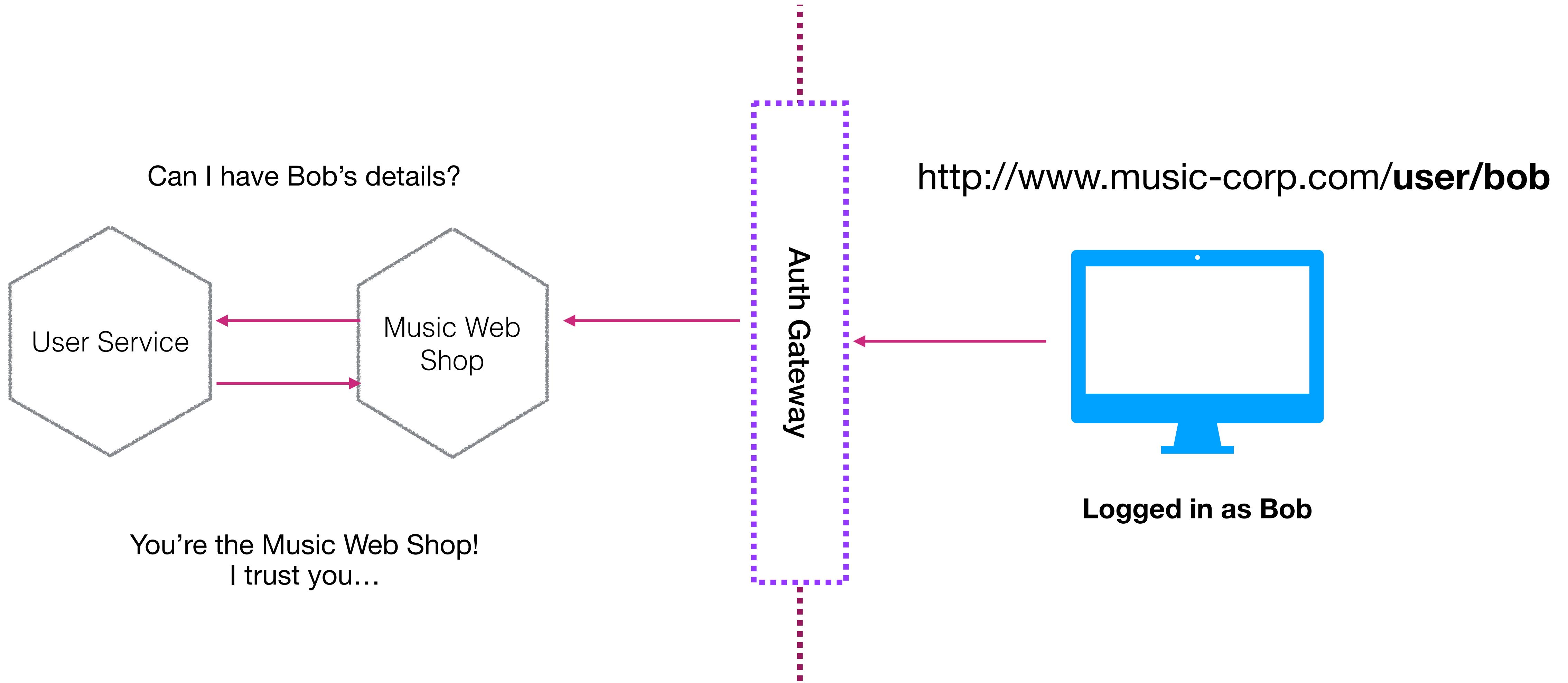
## DOWNSTREAM AUTH - IMPLICIT TRUST?



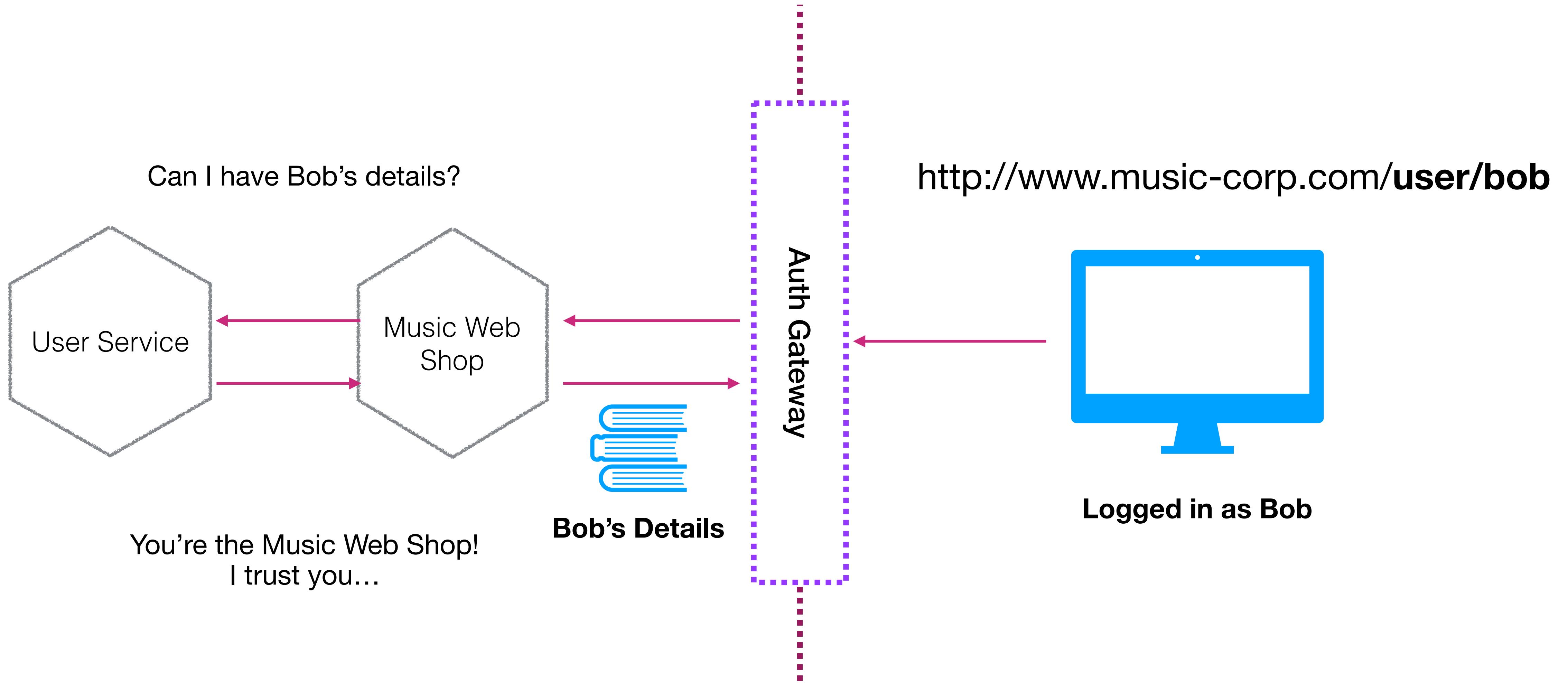
## DOWNSTREAM AUTH - IMPLICIT TRUST?



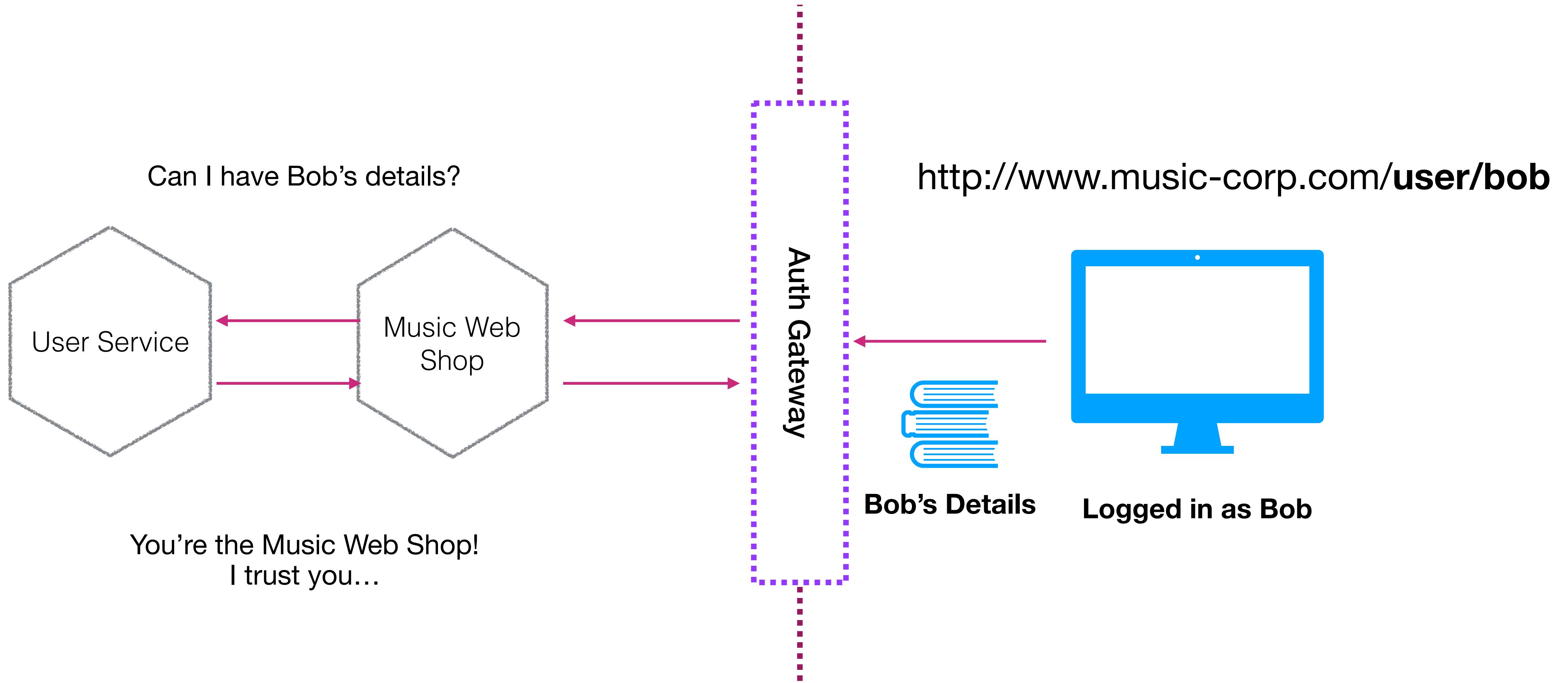
## DOWNSTREAM AUTH - IMPLICIT TRUST?



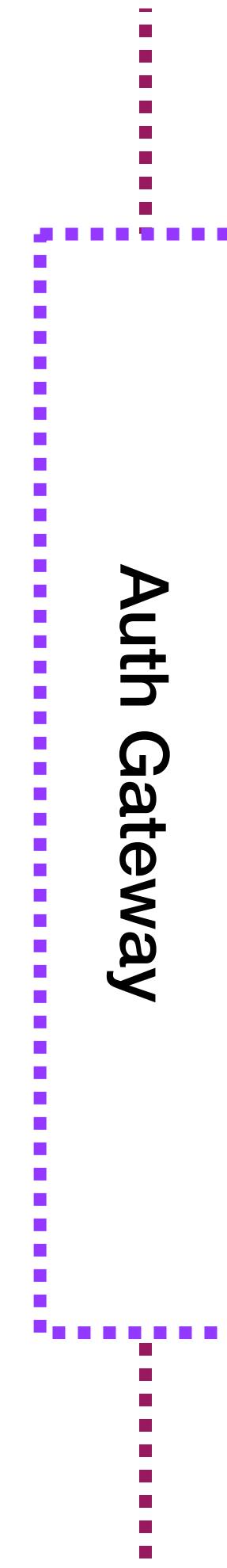
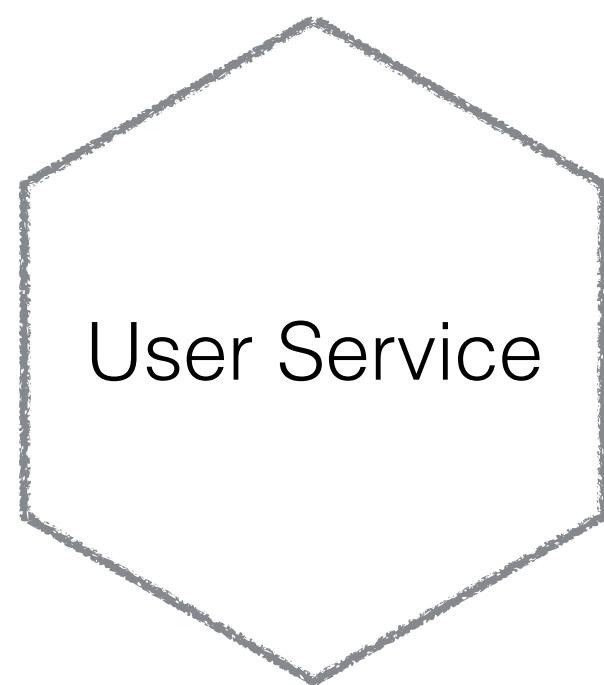
## DOWNSTREAM AUTH - IMPLICIT TRUST?



## DOWNSTREAM AUTH - IMPLICIT TRUST?



# DOWNSTREAM AUTH - IMPLICIT TRUST?

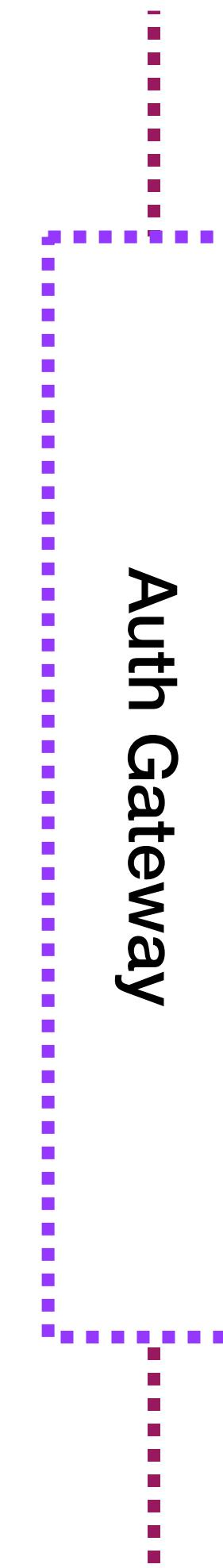
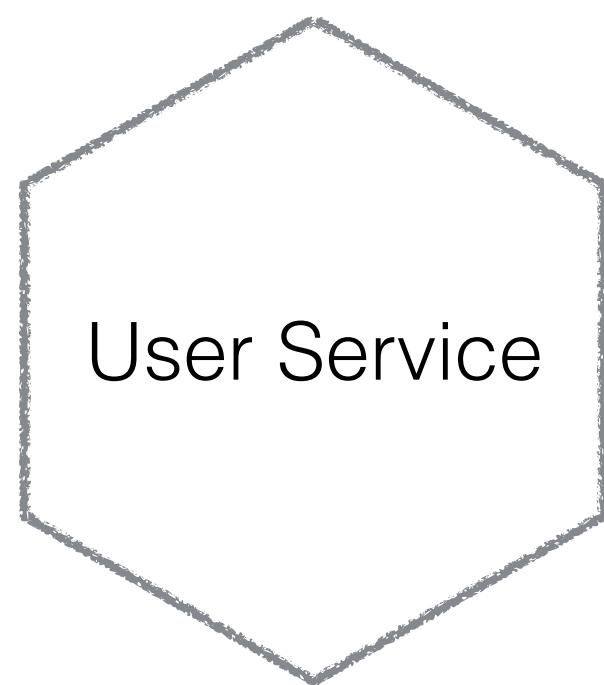


Auth Gateway

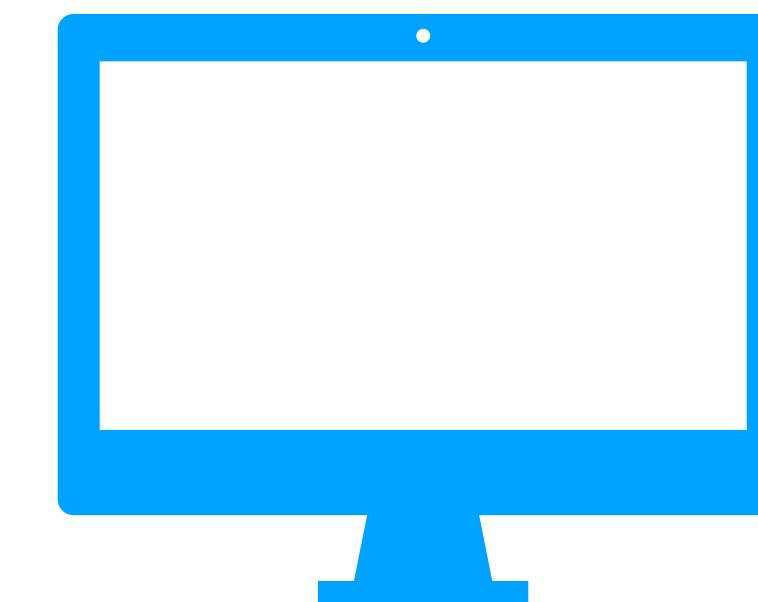


**Logged in as Bob**

## DOWNSTREAM AUTH - IMPLICIT TRUST?

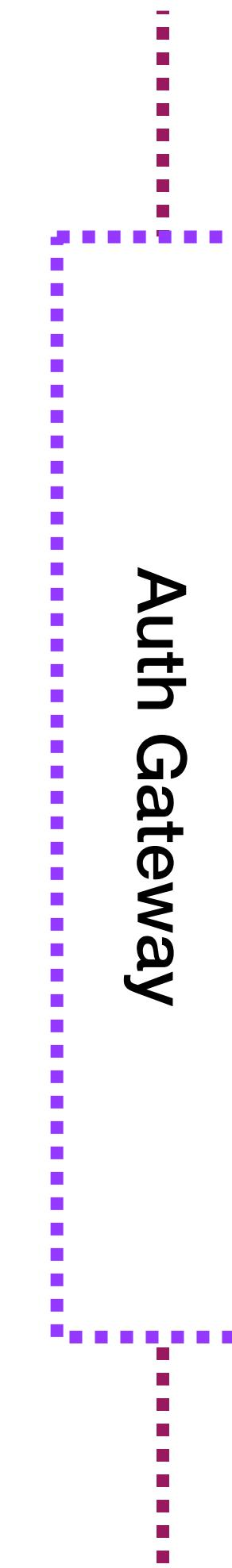
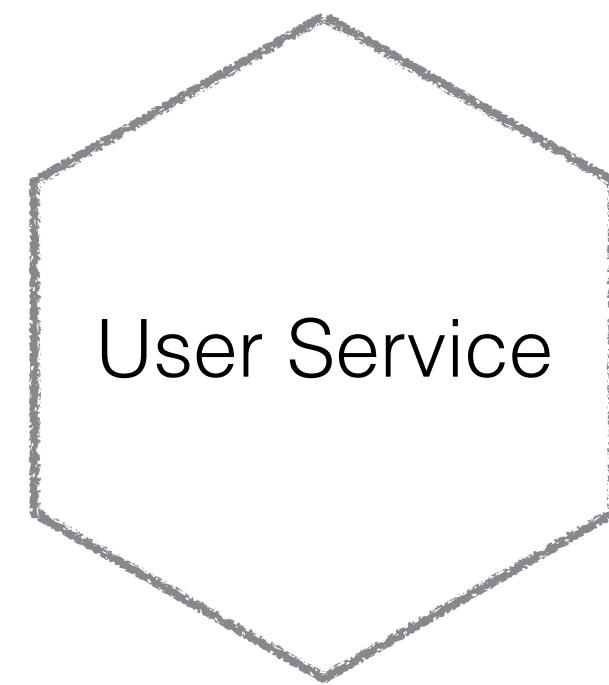


**<http://www.music-corp.com/user/alice>**



**Logged in as Bob**

## DOWNSTREAM AUTH - IMPLICIT TRUST?

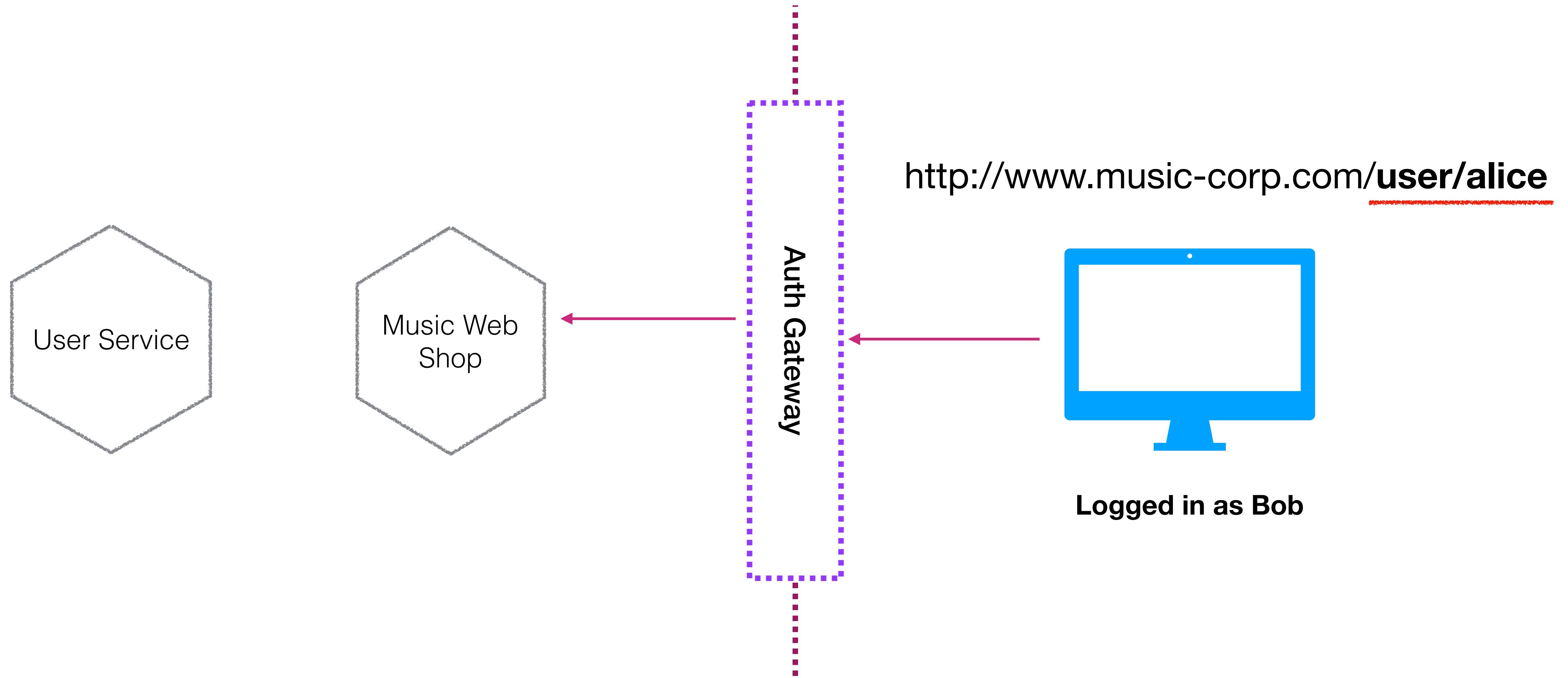


**<http://www.music-corp.com/user/alice>**

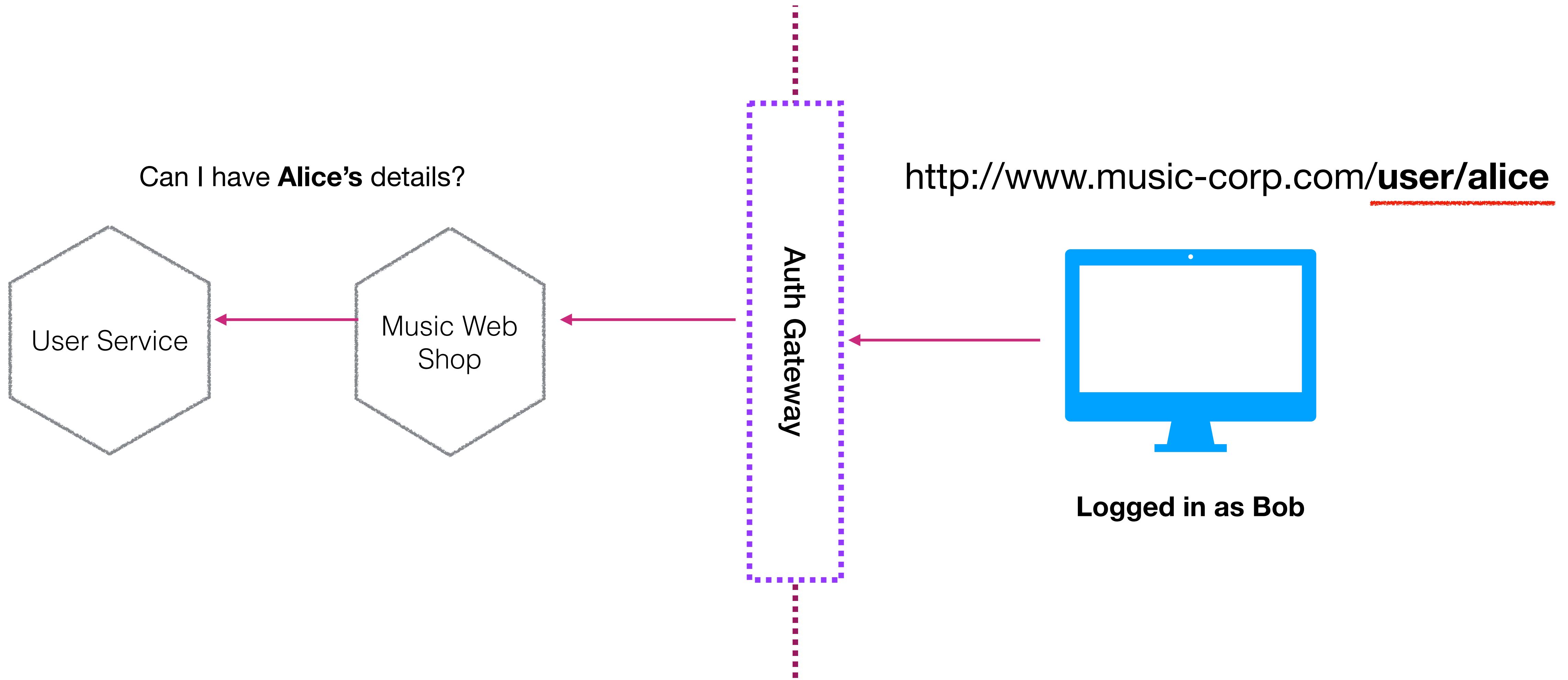


**Logged in as Bob**

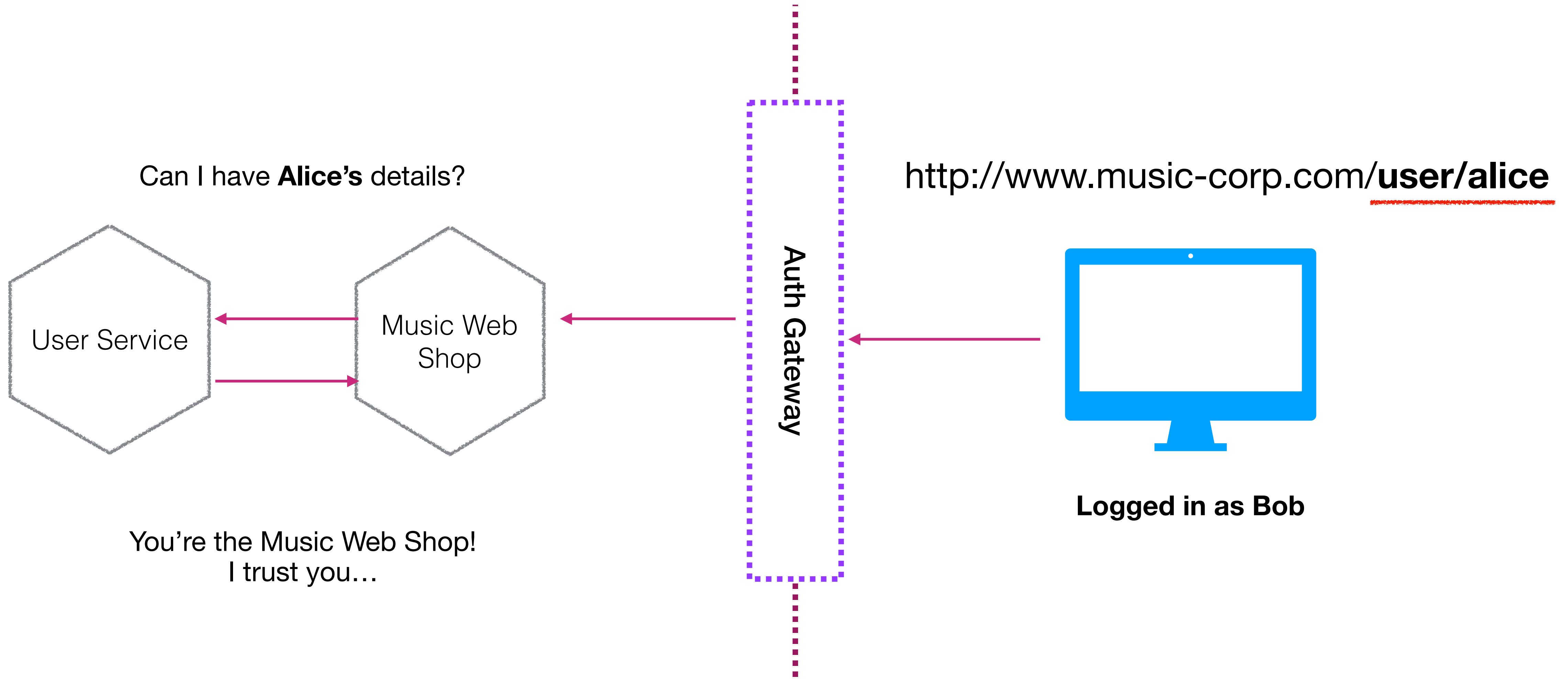
## DOWNSTREAM AUTH - IMPLICIT TRUST?



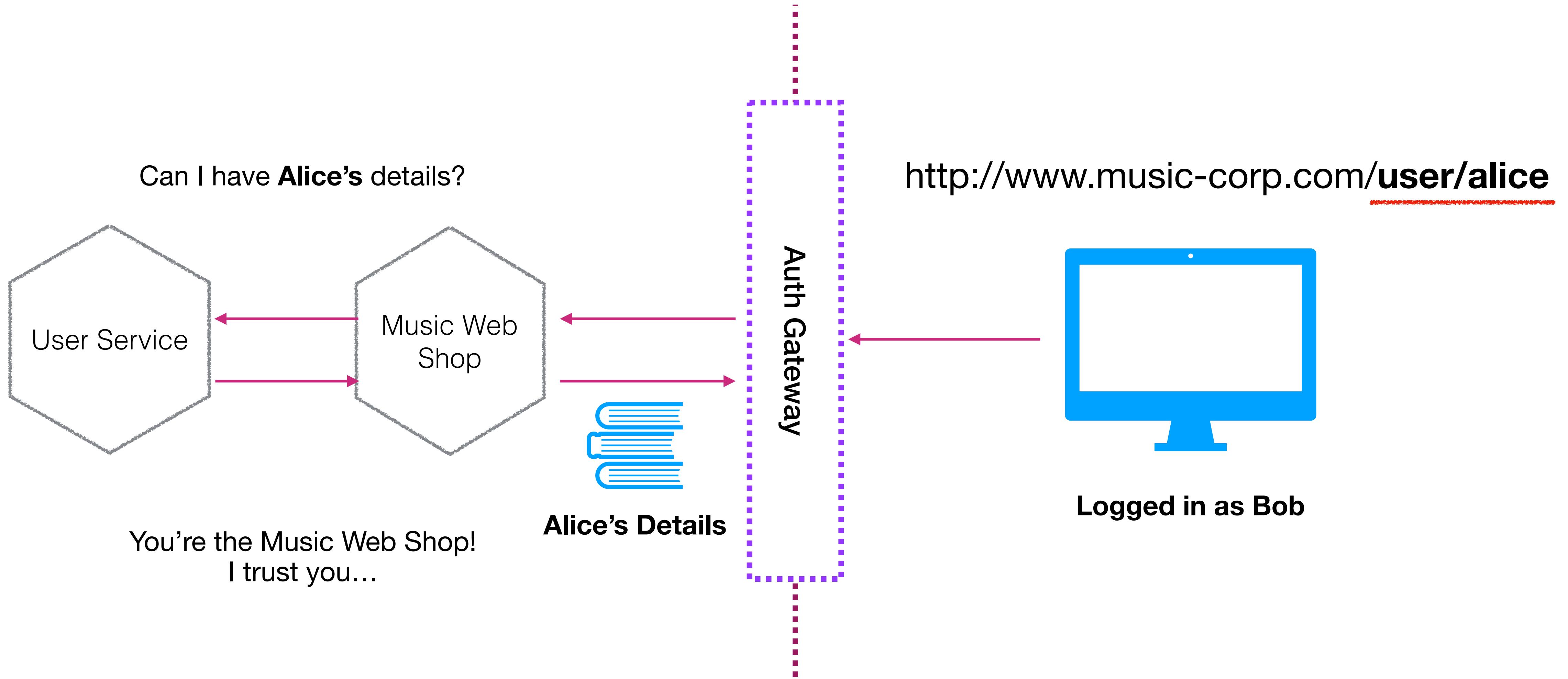
## DOWNSTREAM AUTH - IMPLICIT TRUST?



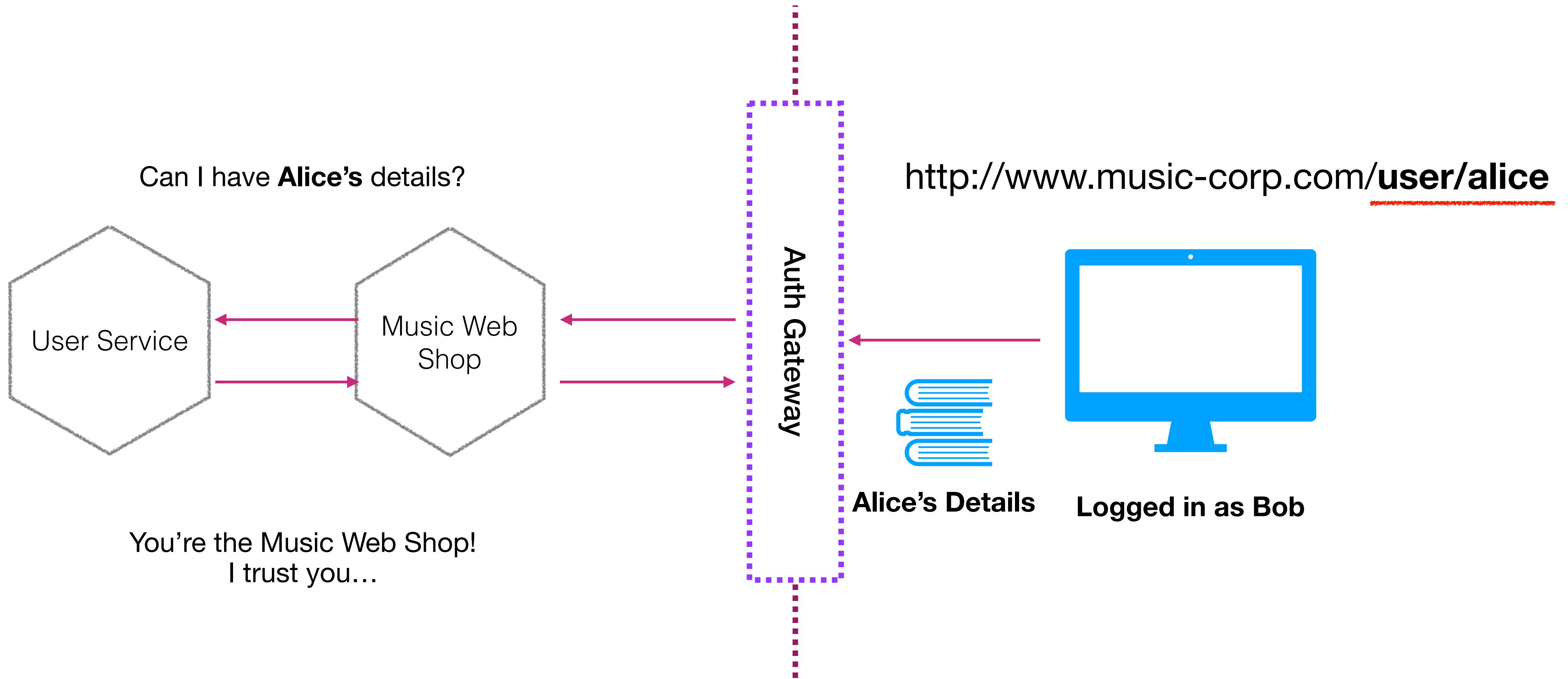
## DOWNSTREAM AUTH - IMPLICIT TRUST?



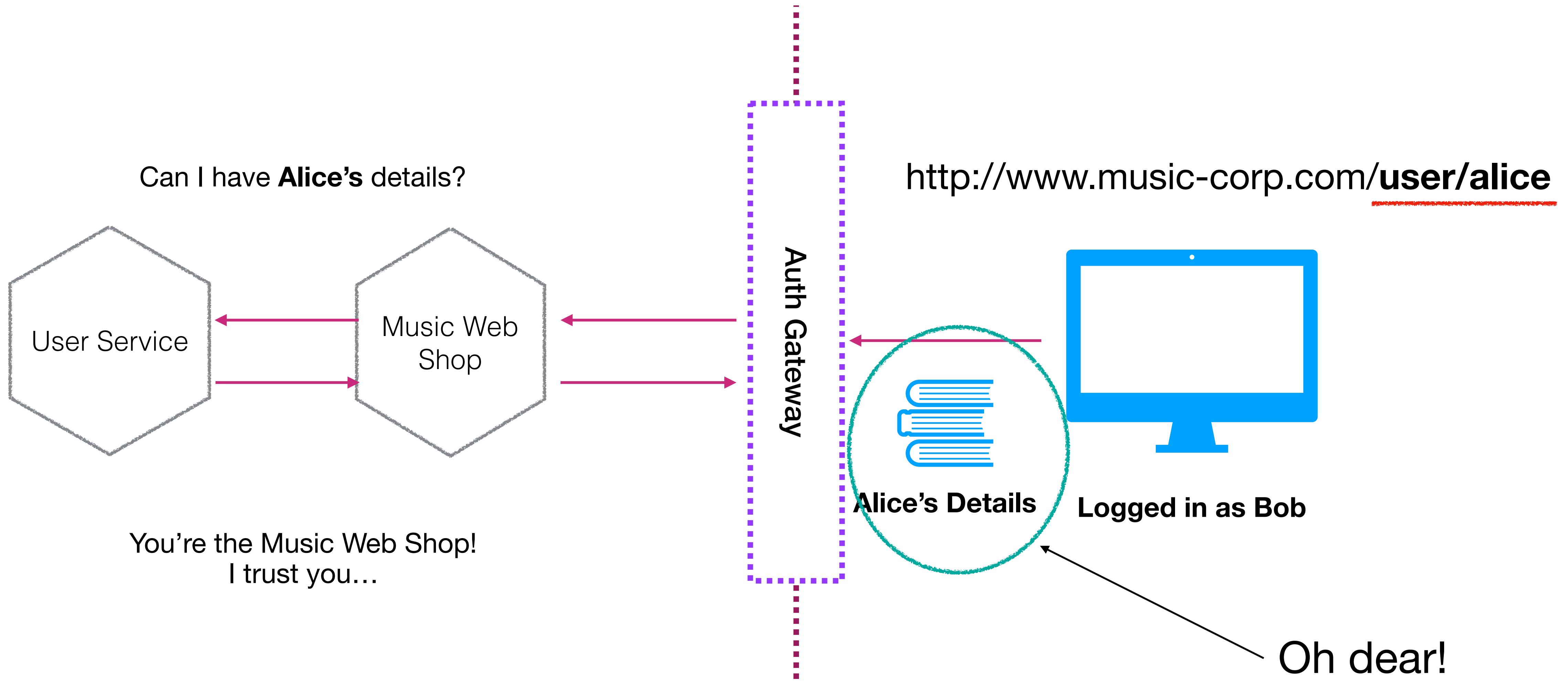
## DOWNSTREAM AUTH - IMPLICIT TRUST?



## DOWNSTREAM AUTH - IMPLICIT TRUST?



## DOWNSTREAM AUTH - IMPLICIT TRUST?



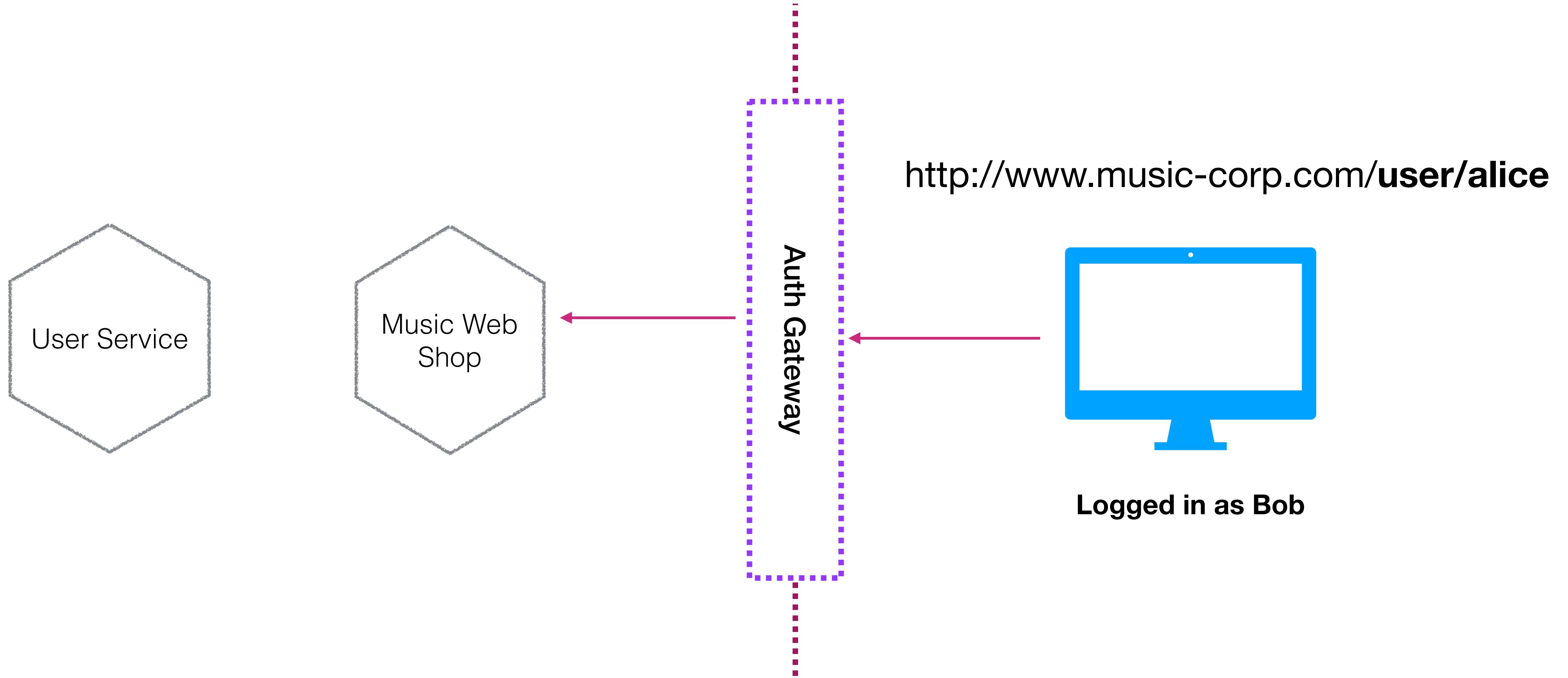


Confused  
Deputy  
Problem!

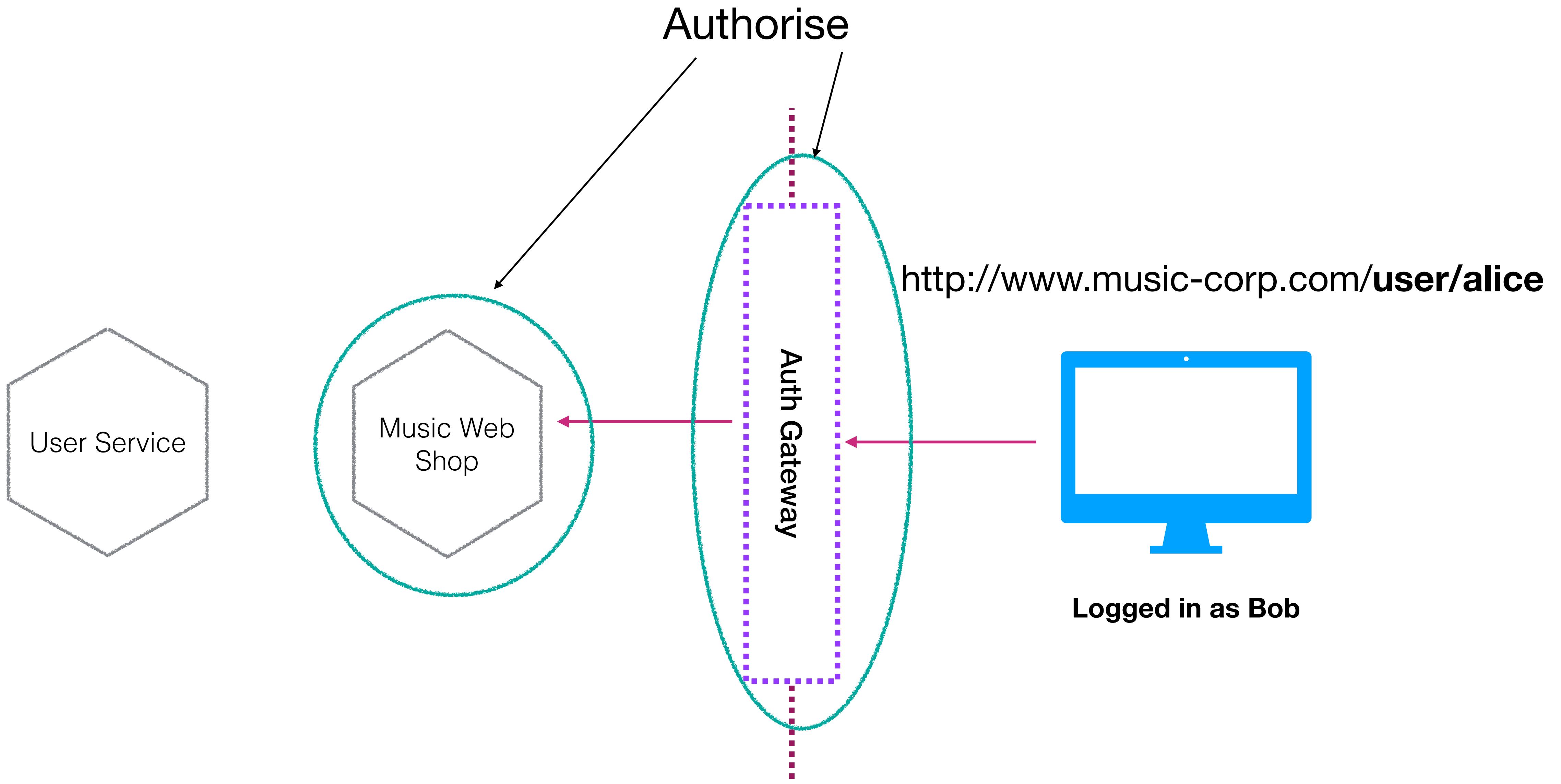
© 2014  
Dave Lundy  
[lundyd@charter.net](mailto:lundyd@charter.net)

<https://www.flickr.com/photos/lundyd/14481829564>

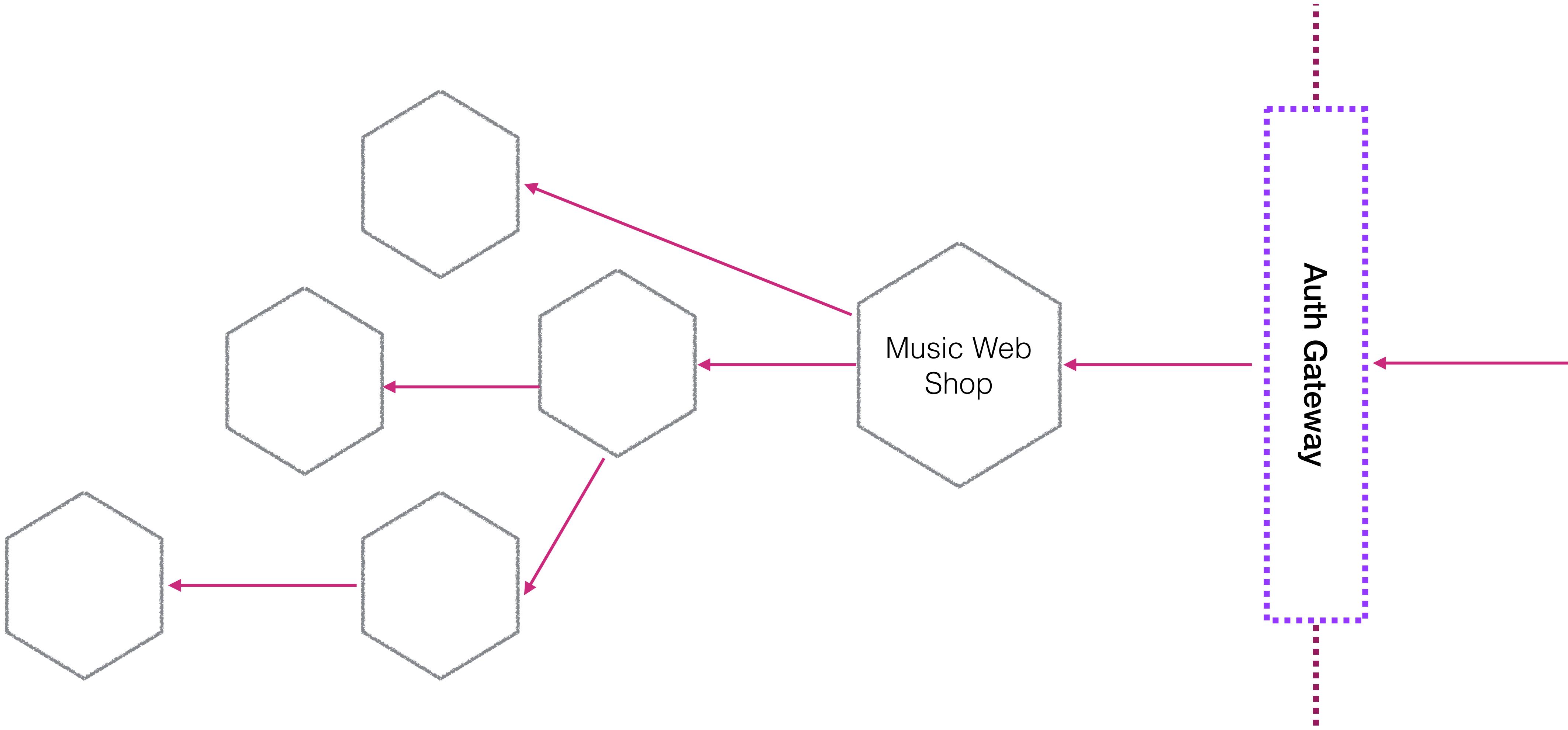
## AUTHORISE UPSTREAM?



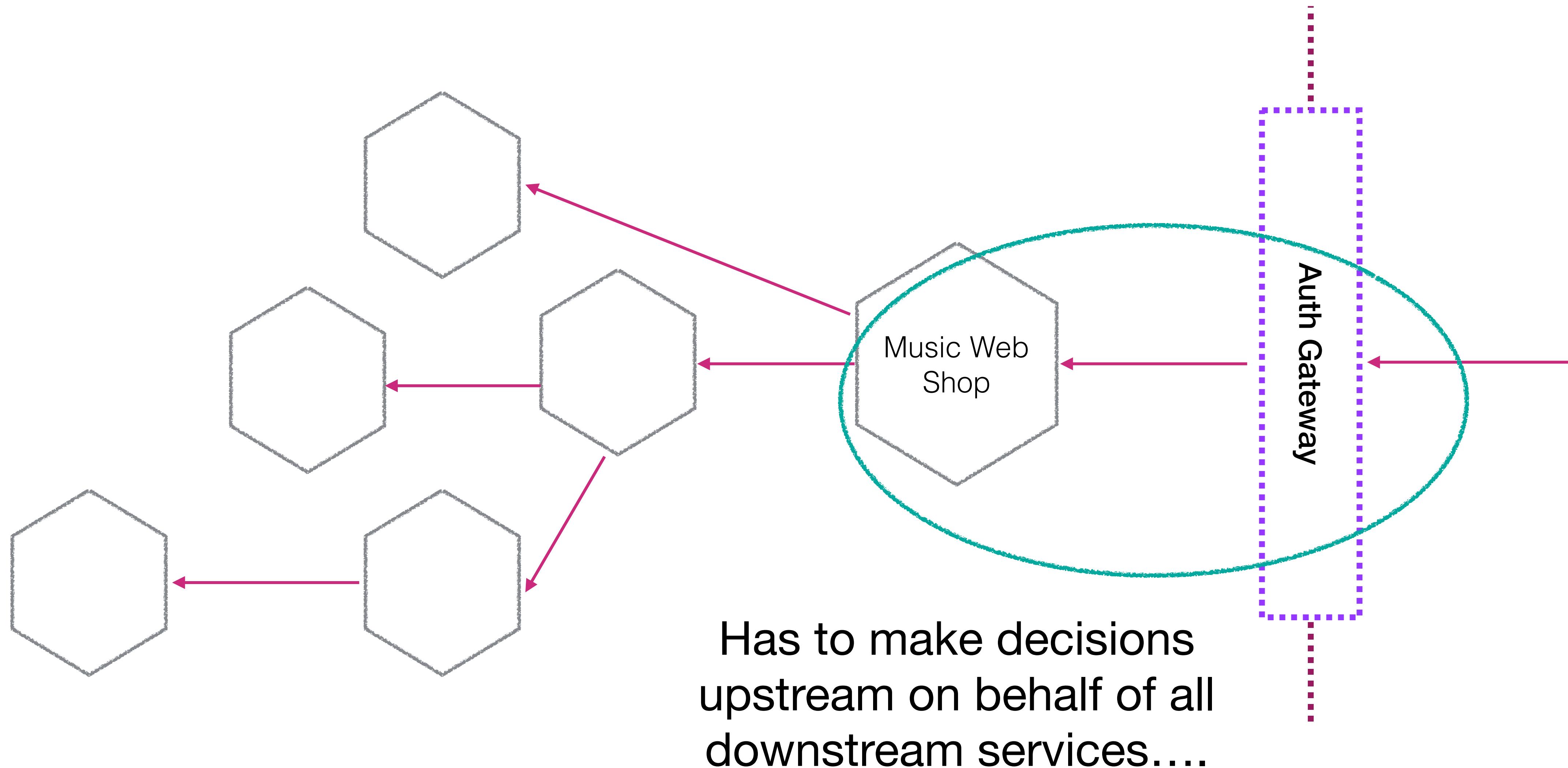
## AUTHORISE UPSTREAM?



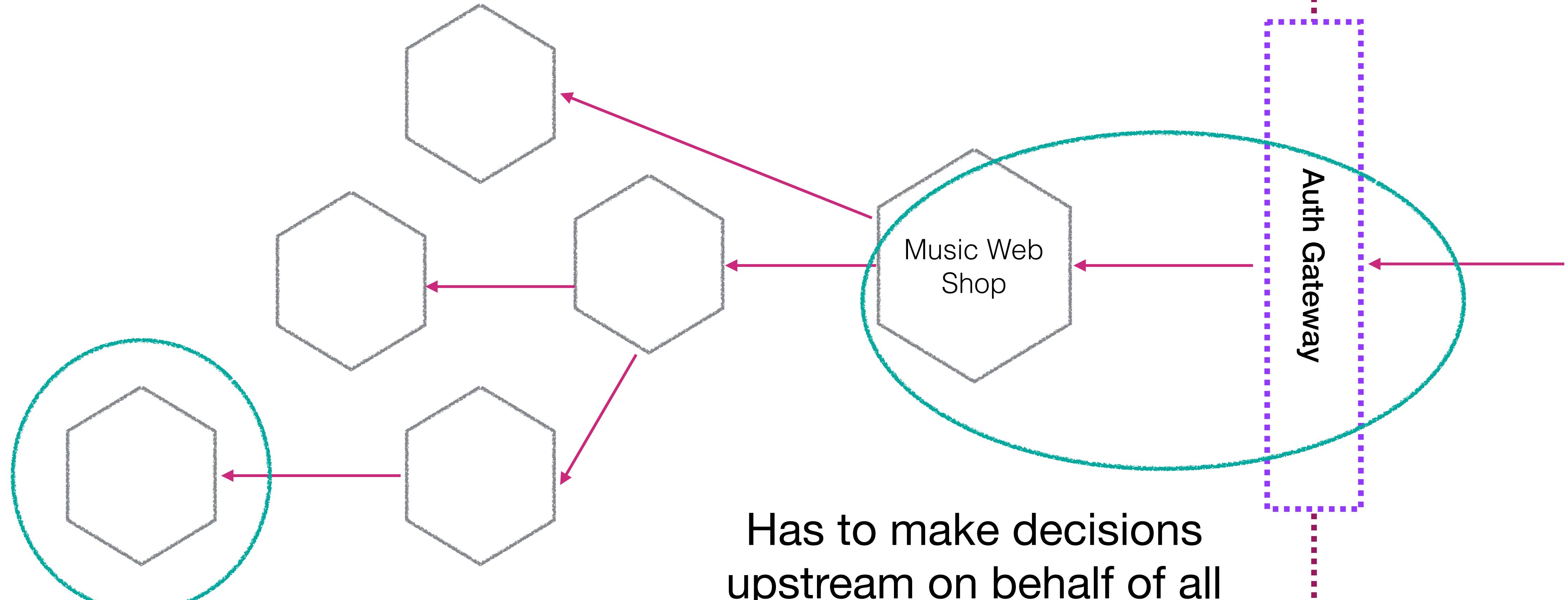
# WHERE DO THE SMARTS LIVE?



# WHERE DO THE SMARTS LIVE?



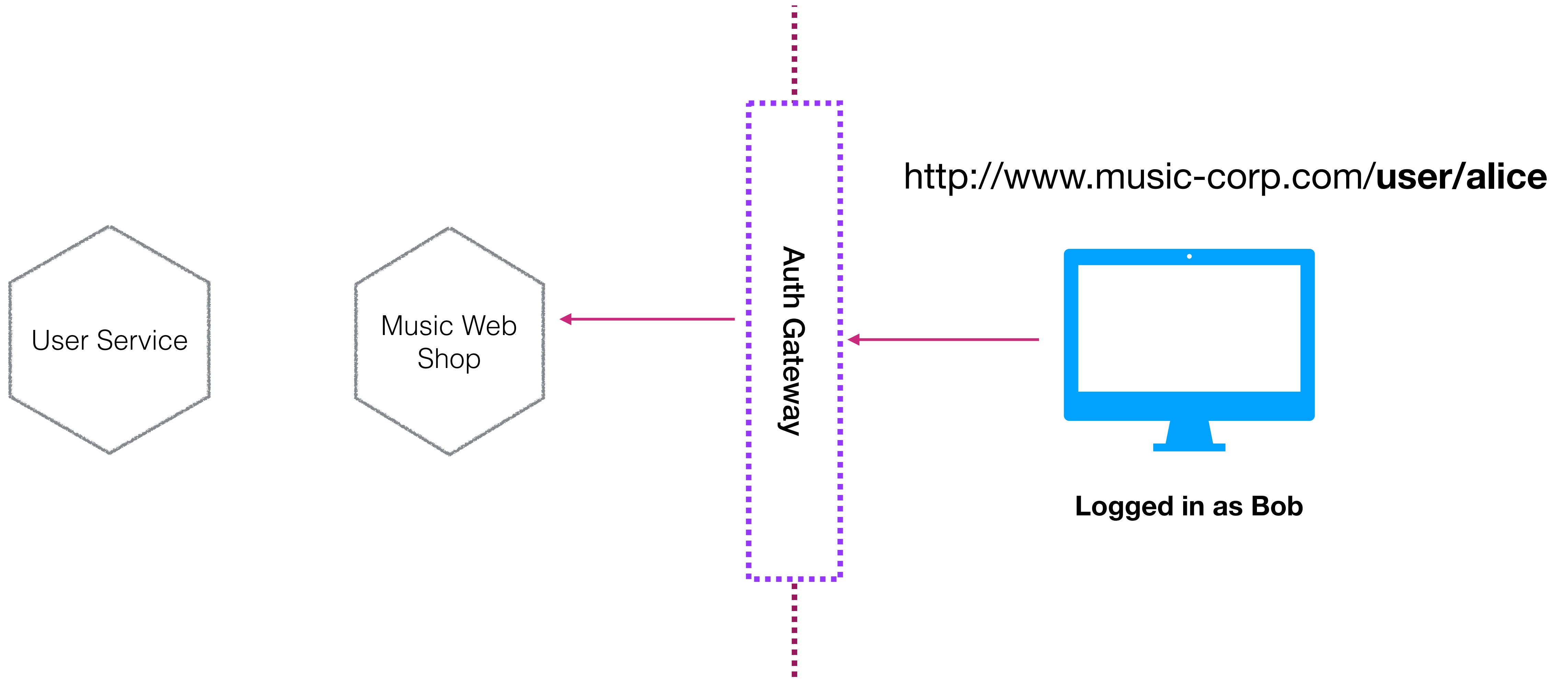
## WHERE DO THE SMARTS LIVE?



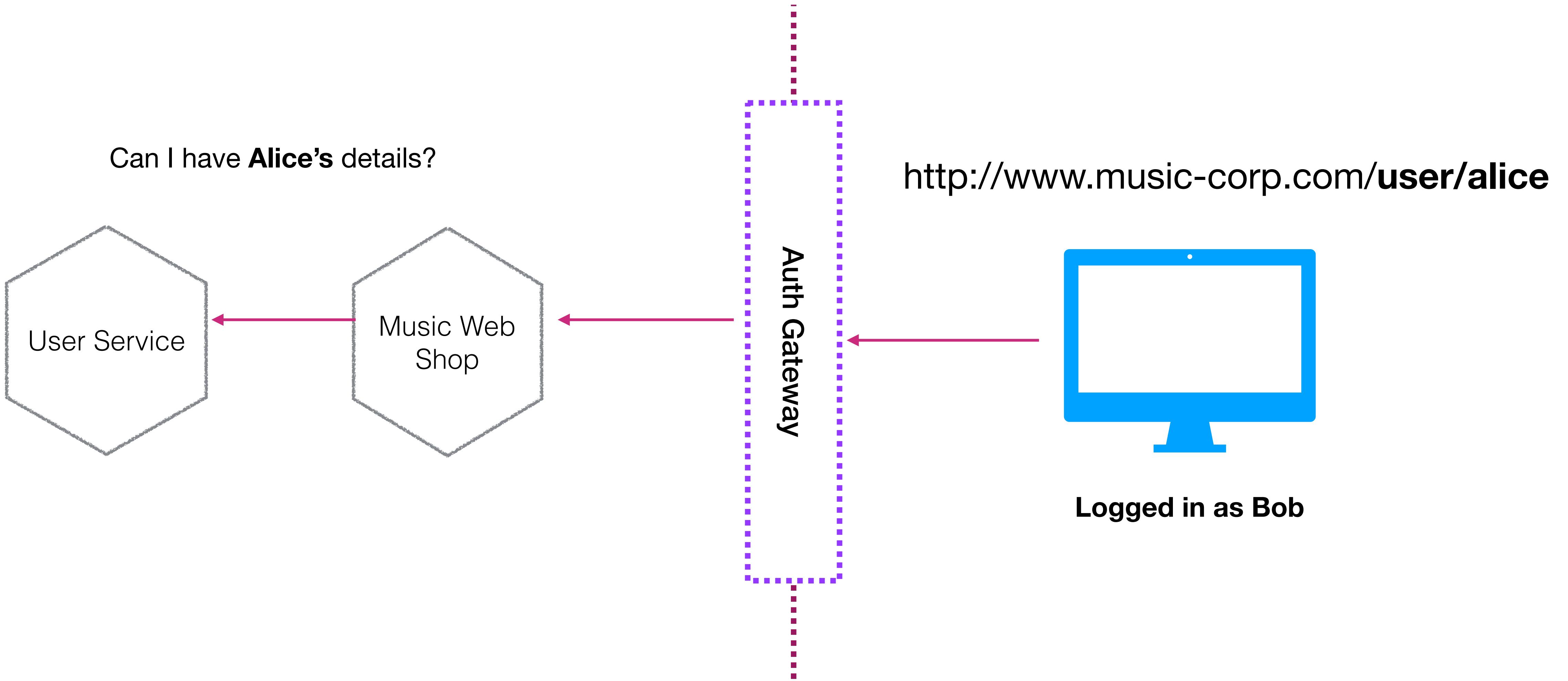
But it can be preferable to push  
this logic to the service itself

Has to make decisions  
upstream on behalf of all  
downstream services....

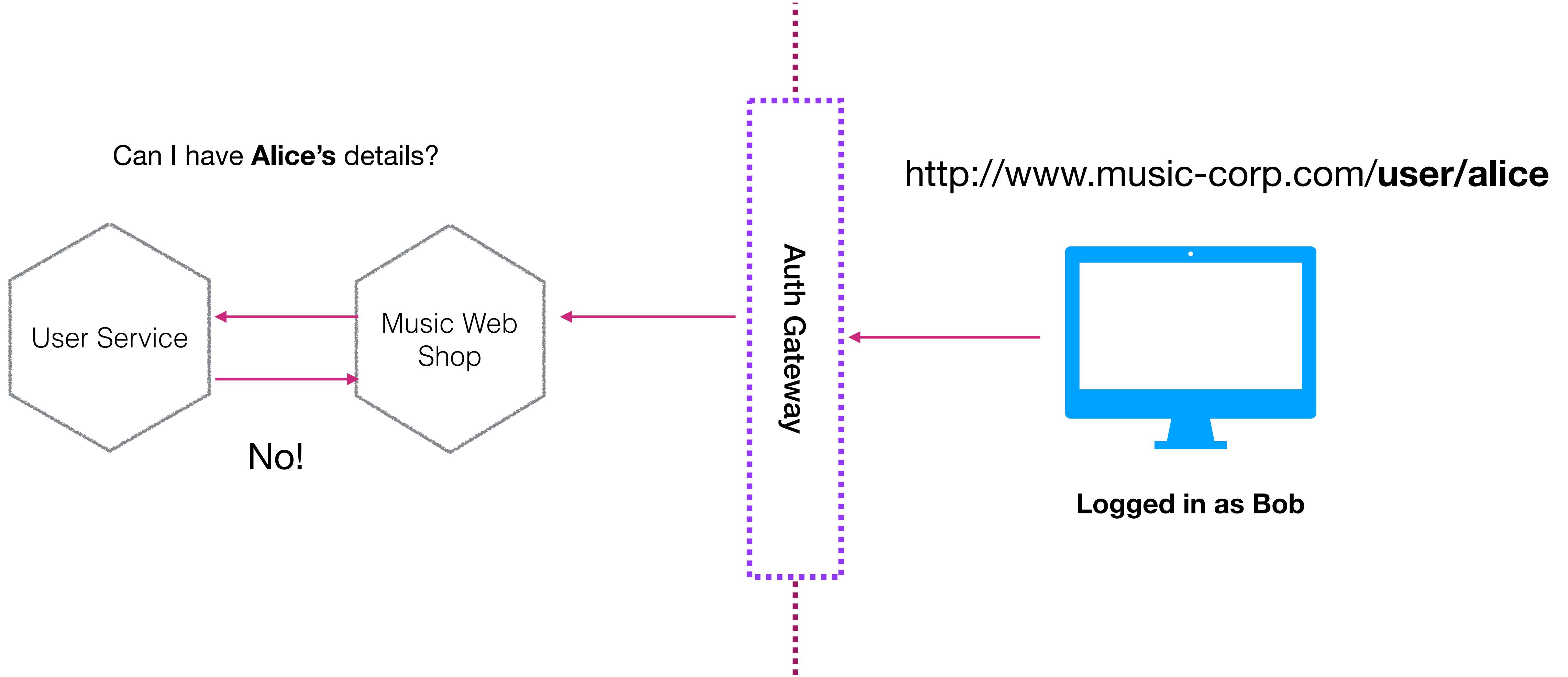
## AUTHORISE DOWNSTREAM



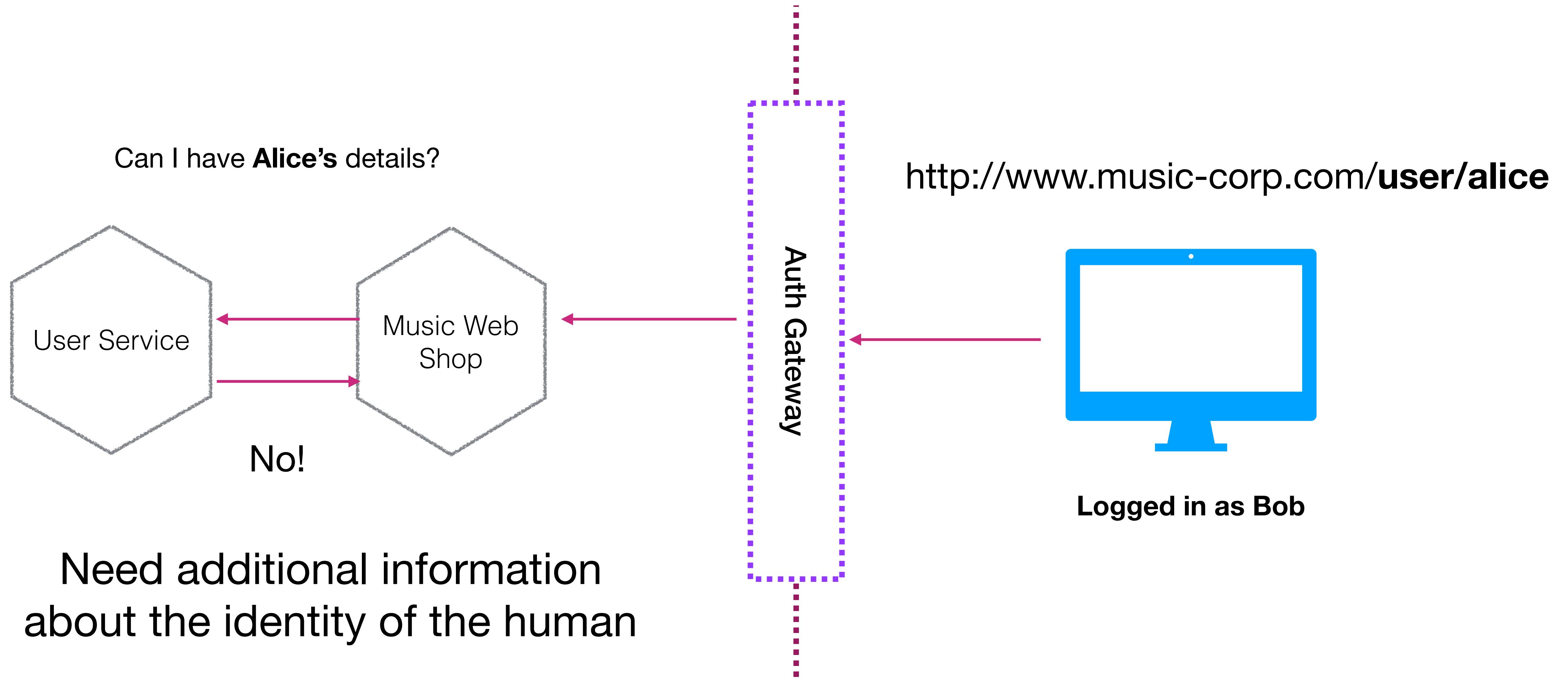
## AUTHORISE DOWNSTREAM



## AUTHORISE DOWNSTREAM



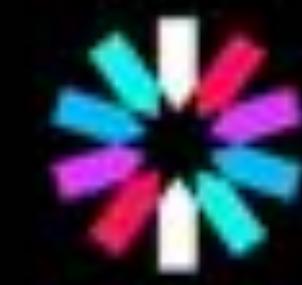
## AUTHORISE DOWNSTREAM





Debugger   Libraries   Introduction   Ask   Get a T-shirt!

Crafted by Auth0



JSON Web Tokens are an open, industry standard [RFC 7519](#) method for representing claims securely between two parties.

JWT.IO allows you to decode, verify and generate JWT.

[LEARN MORE ABOUT JWT](#)

<https://jwt.io/>

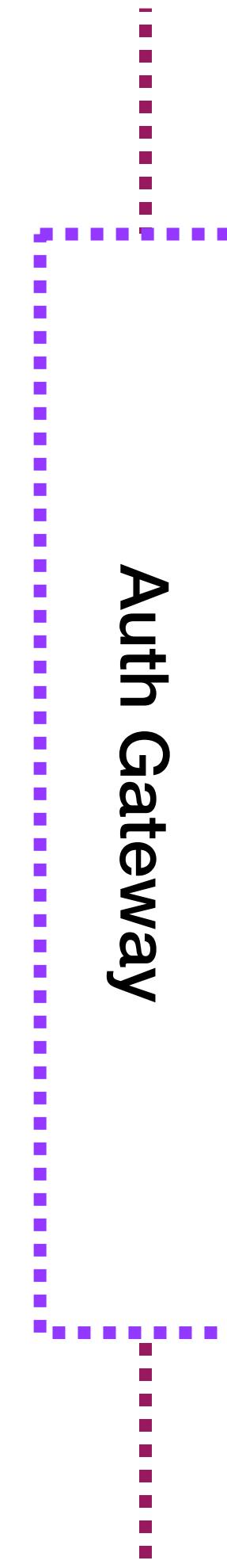
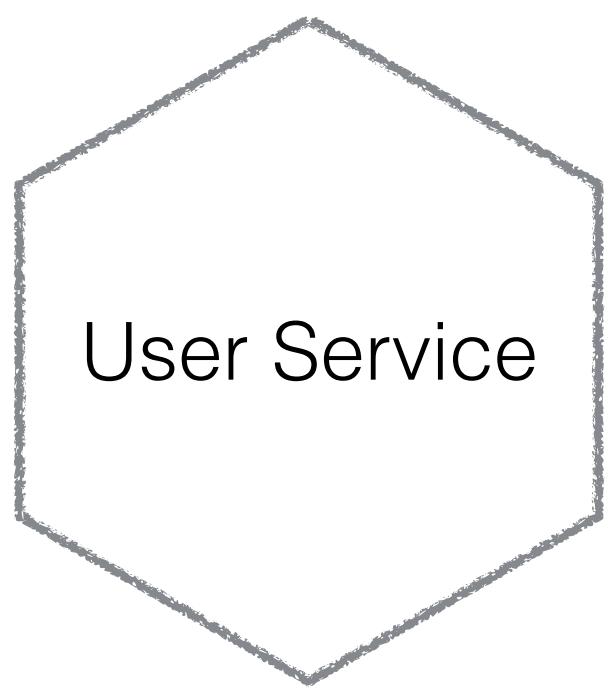
@samnewman

```
{  
  "id": "402ndj39",  
  "name": "Alice Alison"  
}
```

```
{  
  "id": "402ndj39",  
  "name": "Alice Alison"  
}
```

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.  
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4  
gRG9lIiwiaXNTb2NpYWwiOnRydWV9.  
4pcPyMD09o1PSyXnrXCjTwXyr4BsezdI1AVTmud2fU4

# USING JWT TOKENS

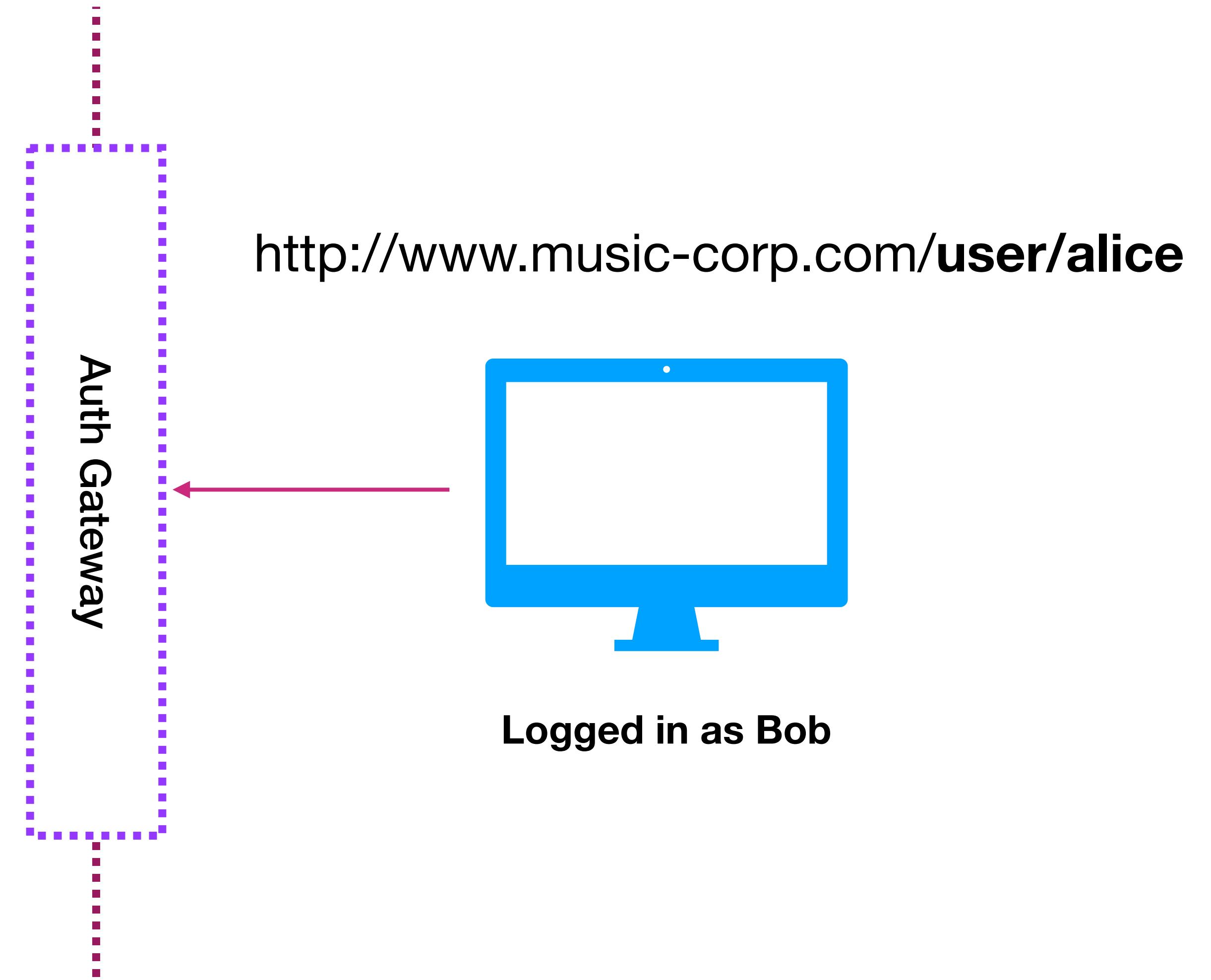
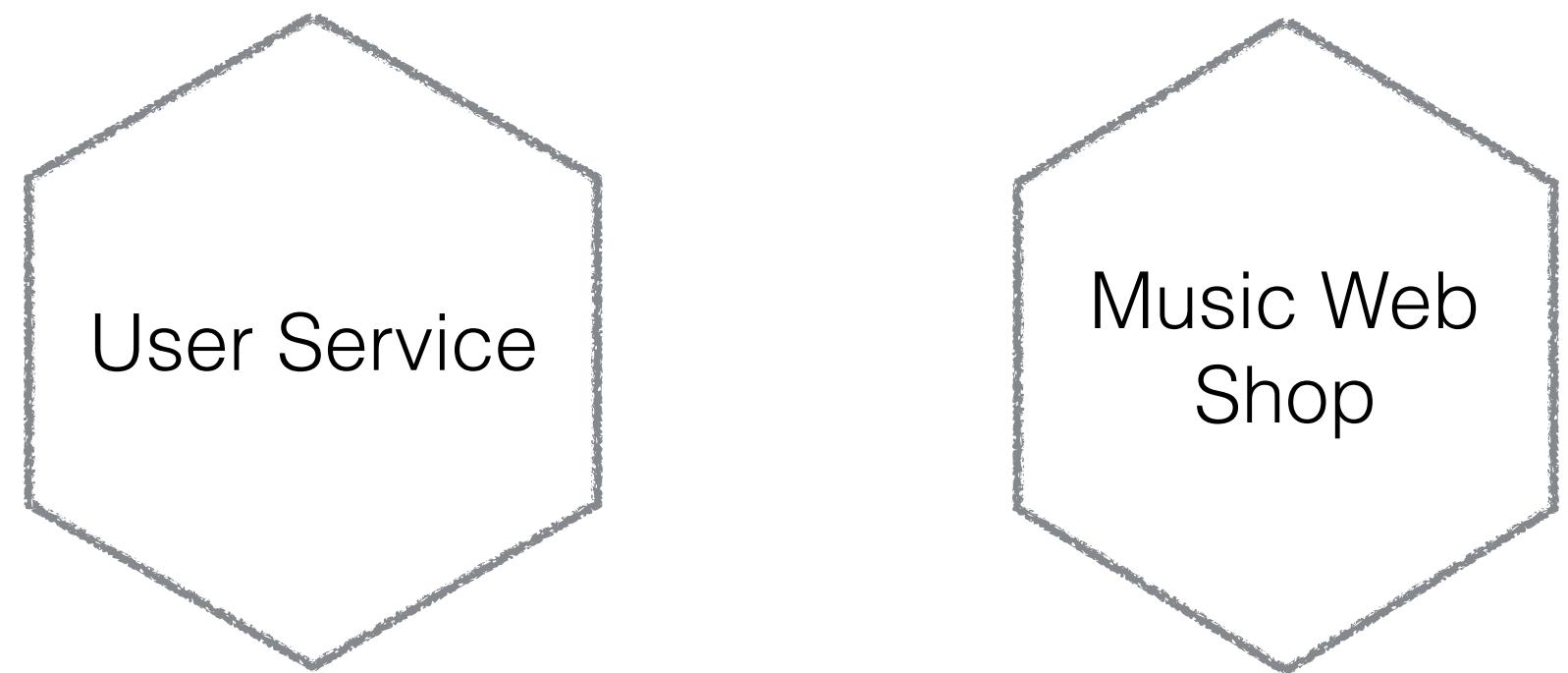


**`http://www.music-corp.com/user/alice`**

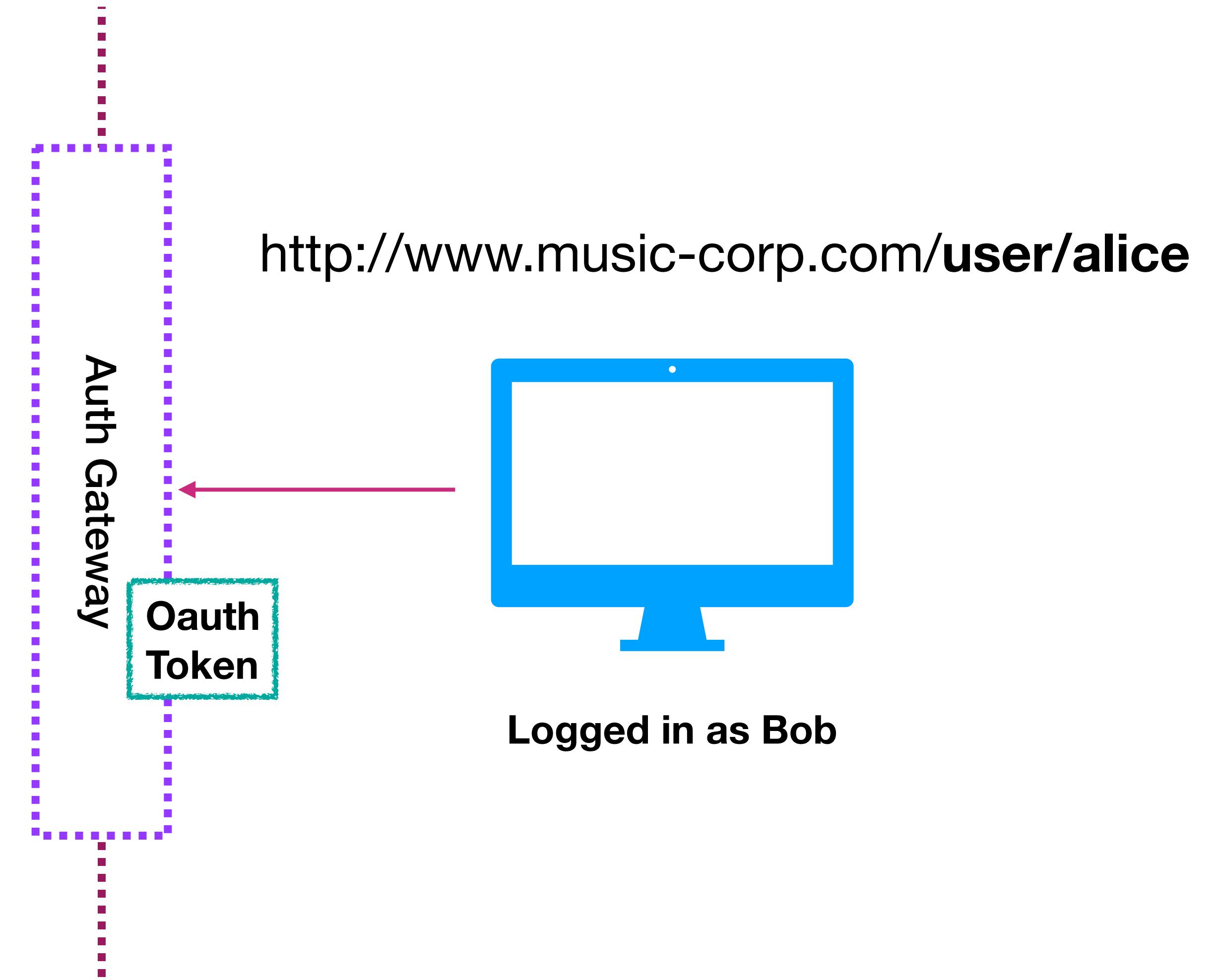
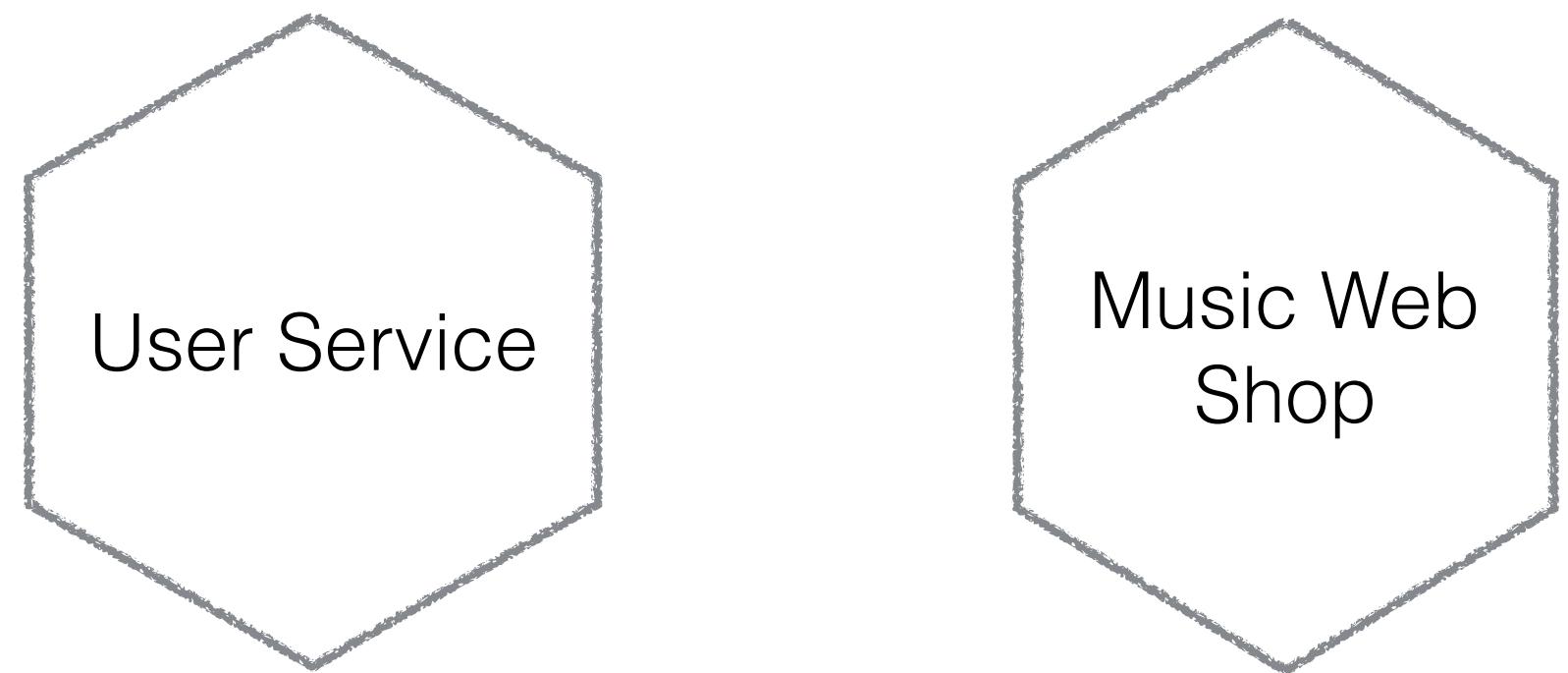


**Logged in as Bob**

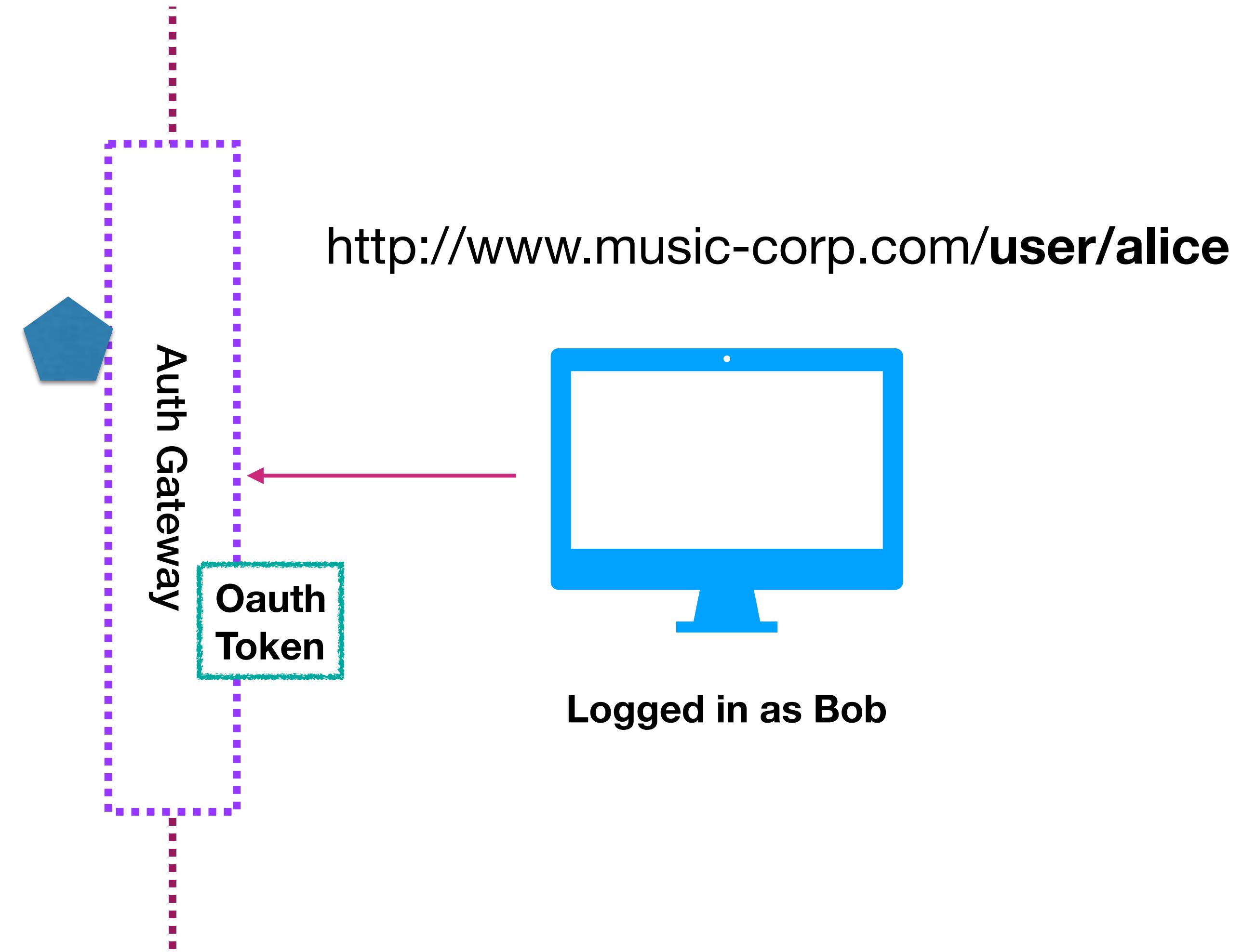
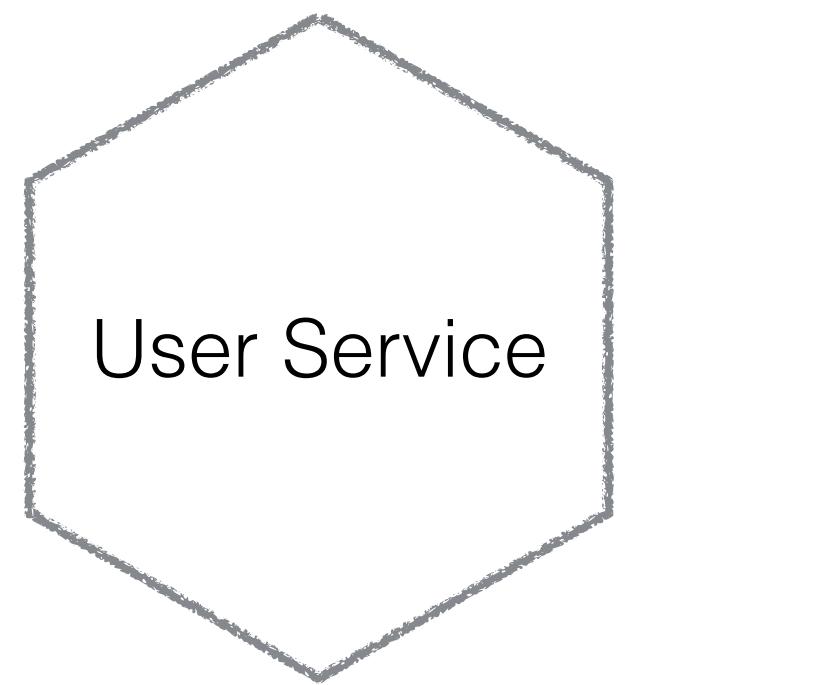
# USING JWT TOKENS



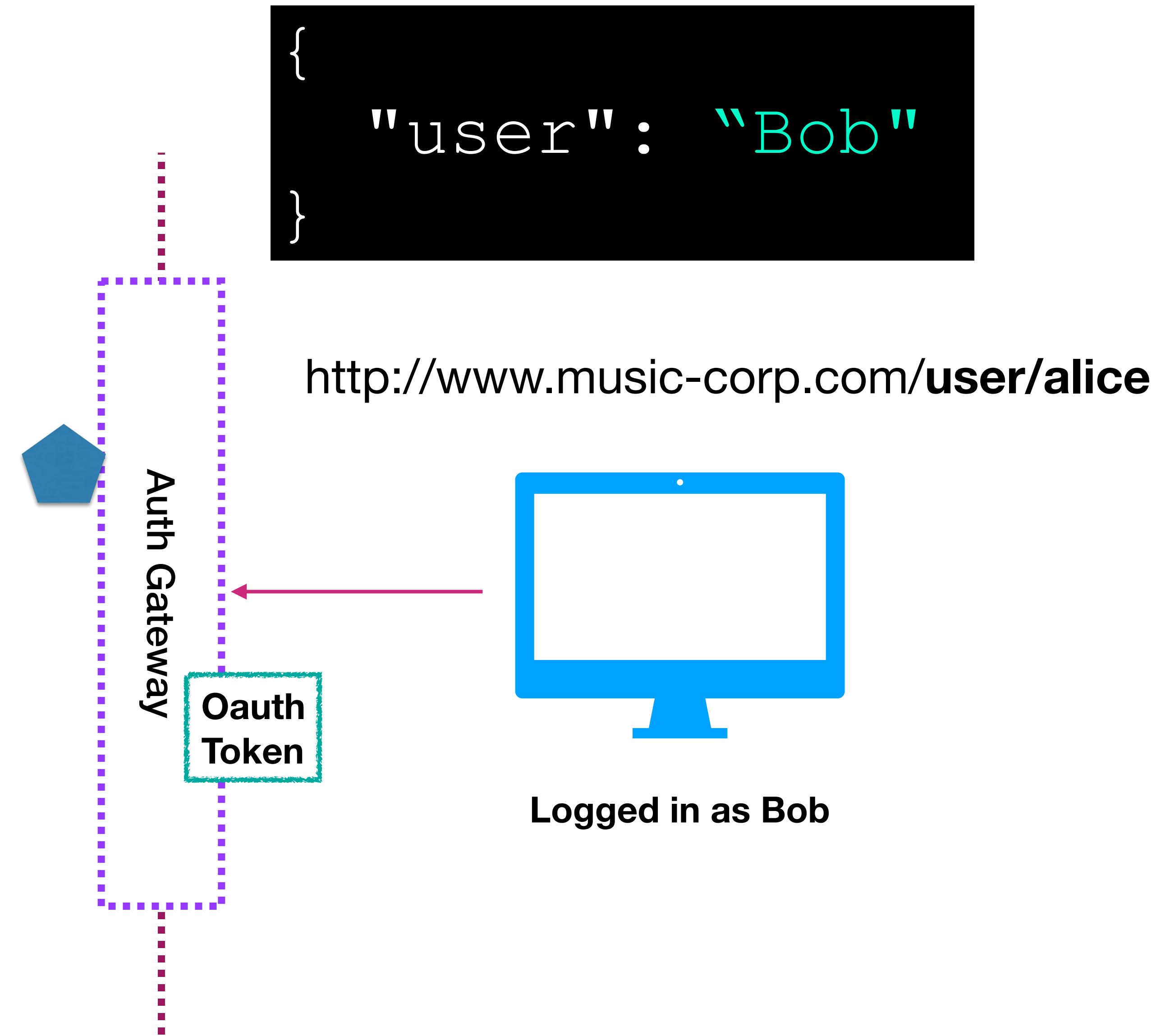
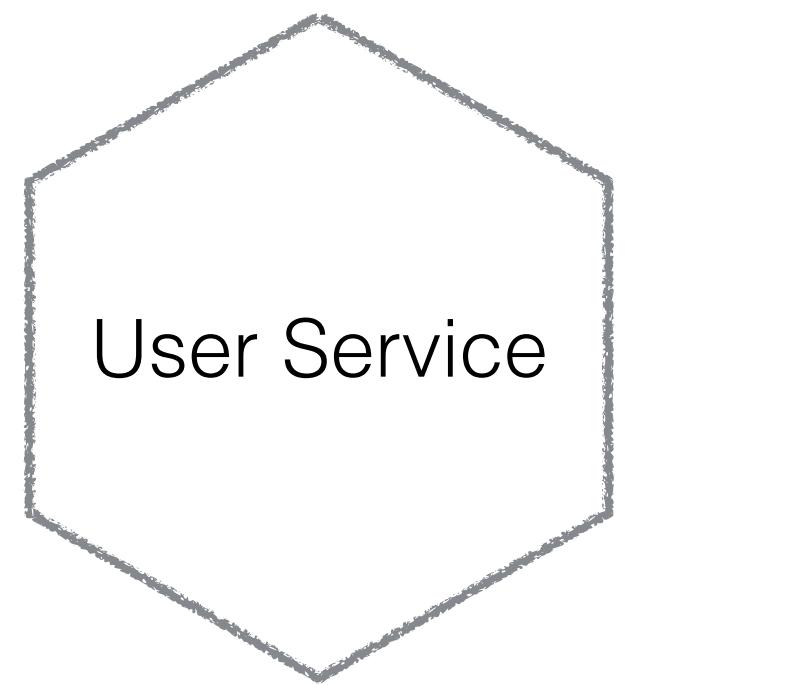
# USING JWT TOKENS



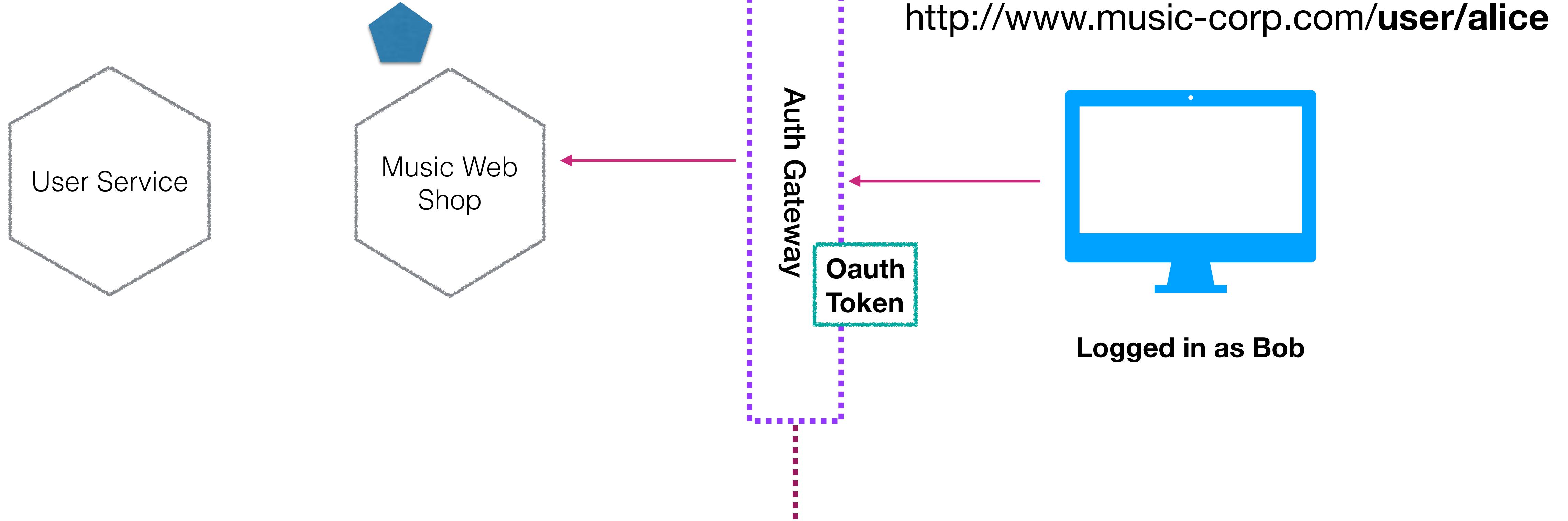
# USING JWT TOKENS



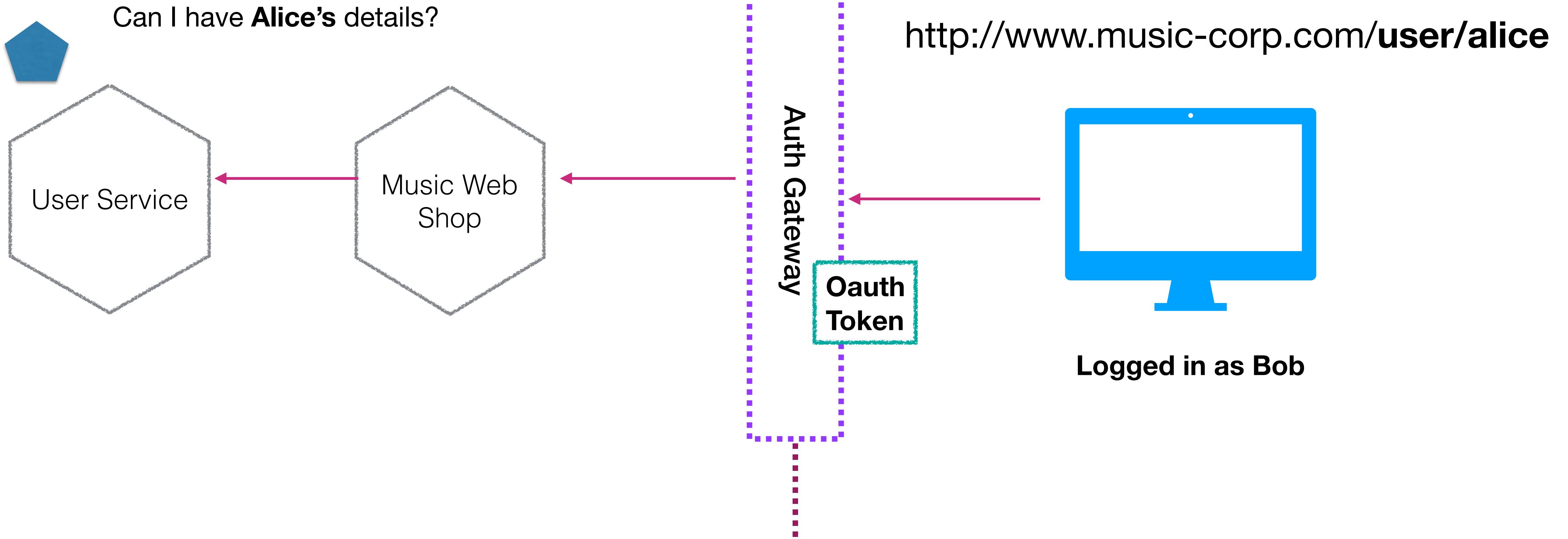
# USING JWT TOKENS



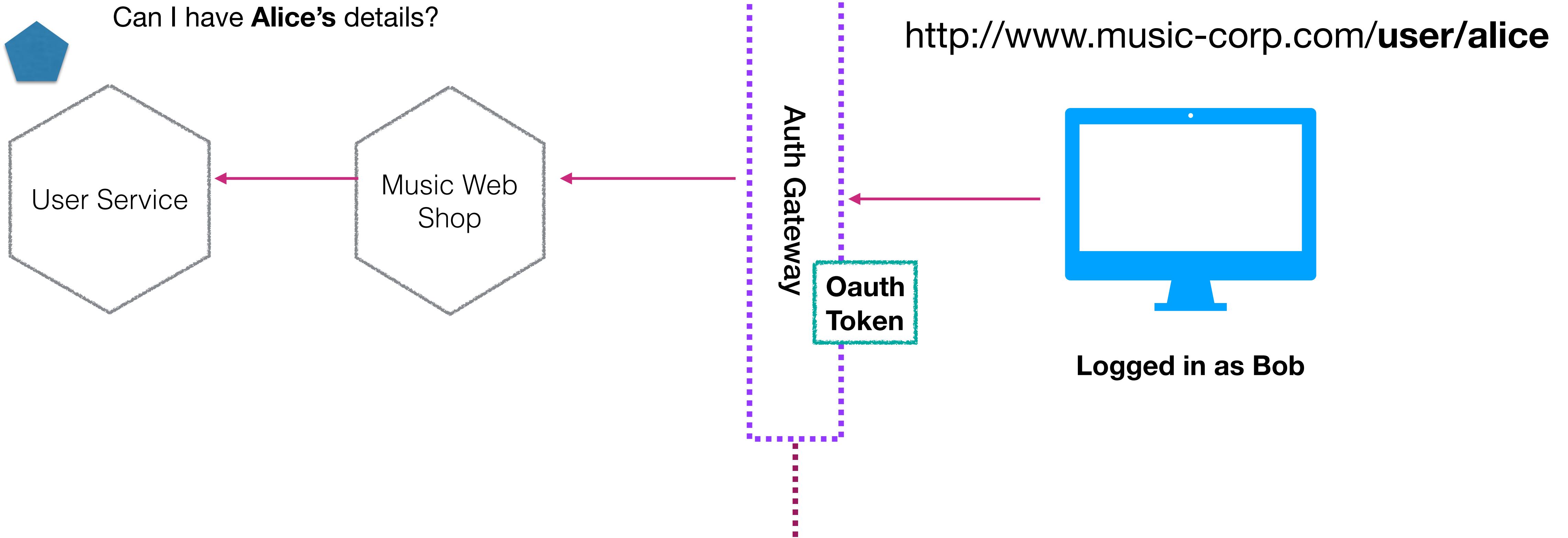
# USING JWT TOKENS



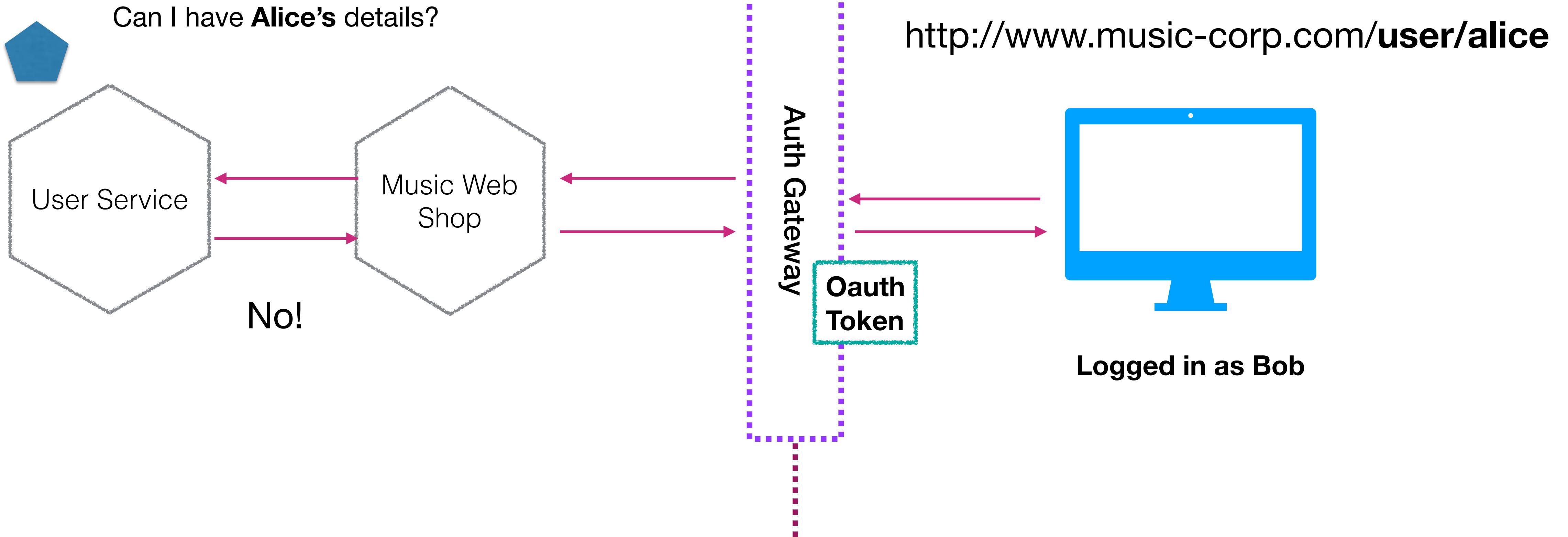
# USING JWT TOKENS



# USING JWT TOKENS



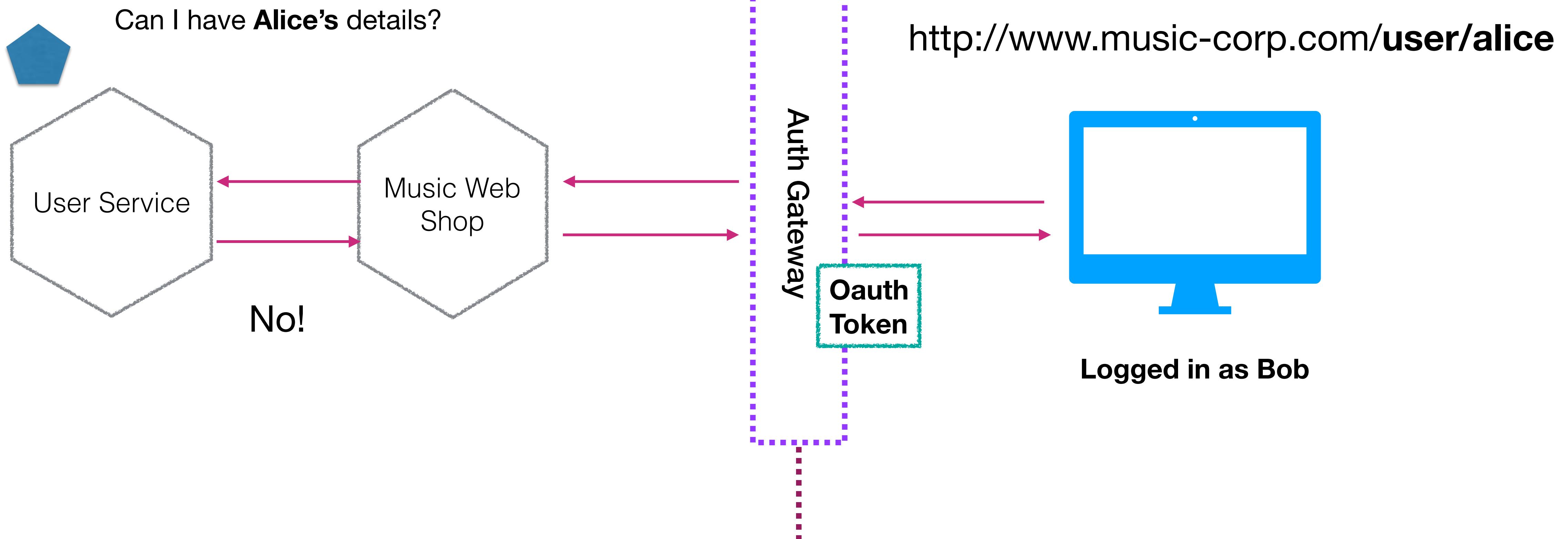
# USING JWT TOKENS



## USING JWT TOKENS

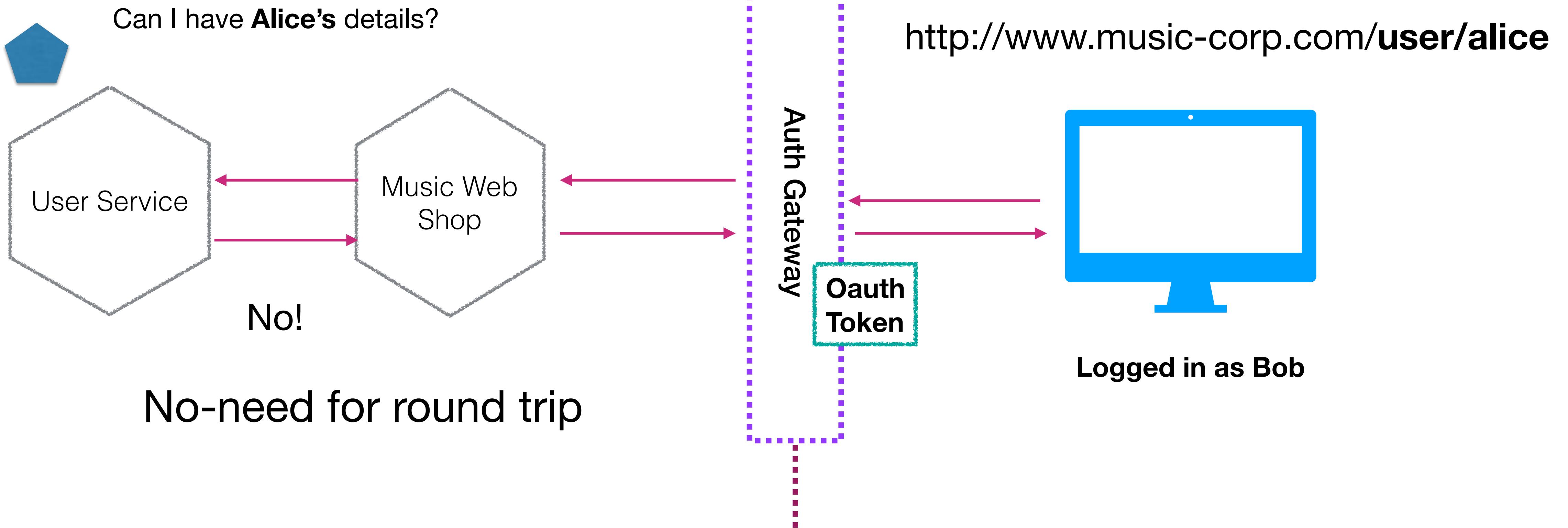
Token can be validated in the user service

```
{  
  "user": "Bob"  
}
```



## USING JWT TOKENS

Token can be validated in the user service



# SERVICE MESHES



**Linkerd**

**<https://linkerd.io>**

# SERVICE MESHES



**Linkerd**

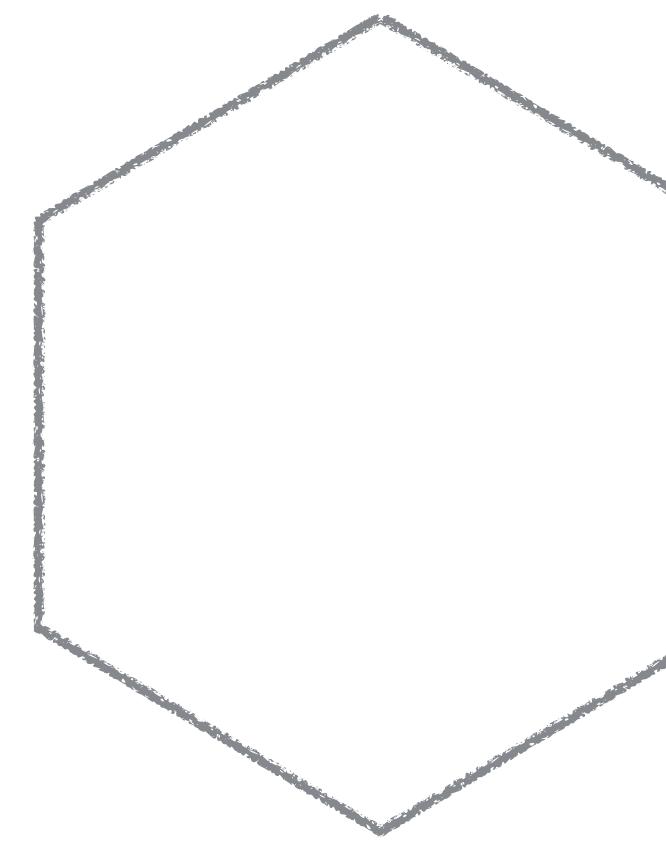
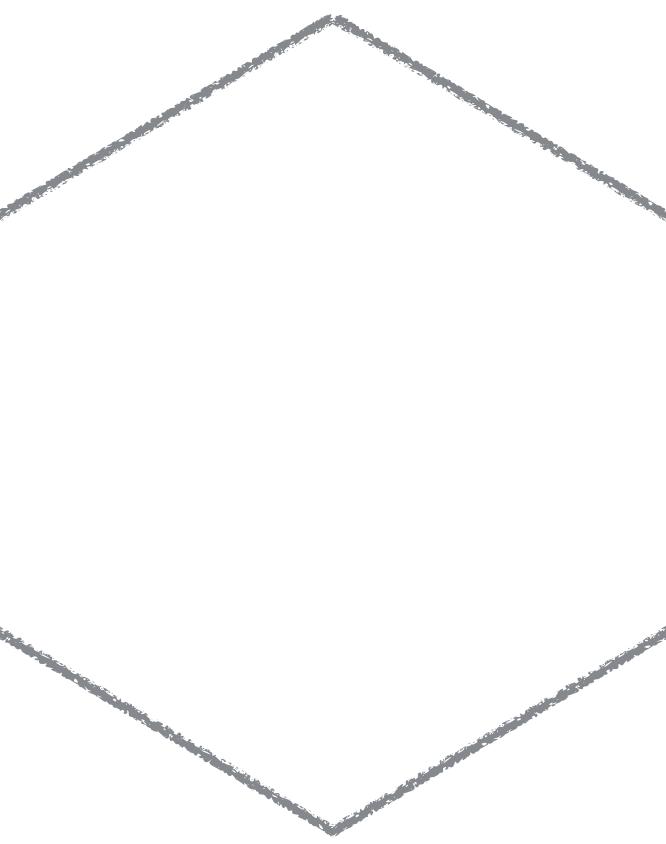
**<https://linkerd.io>**



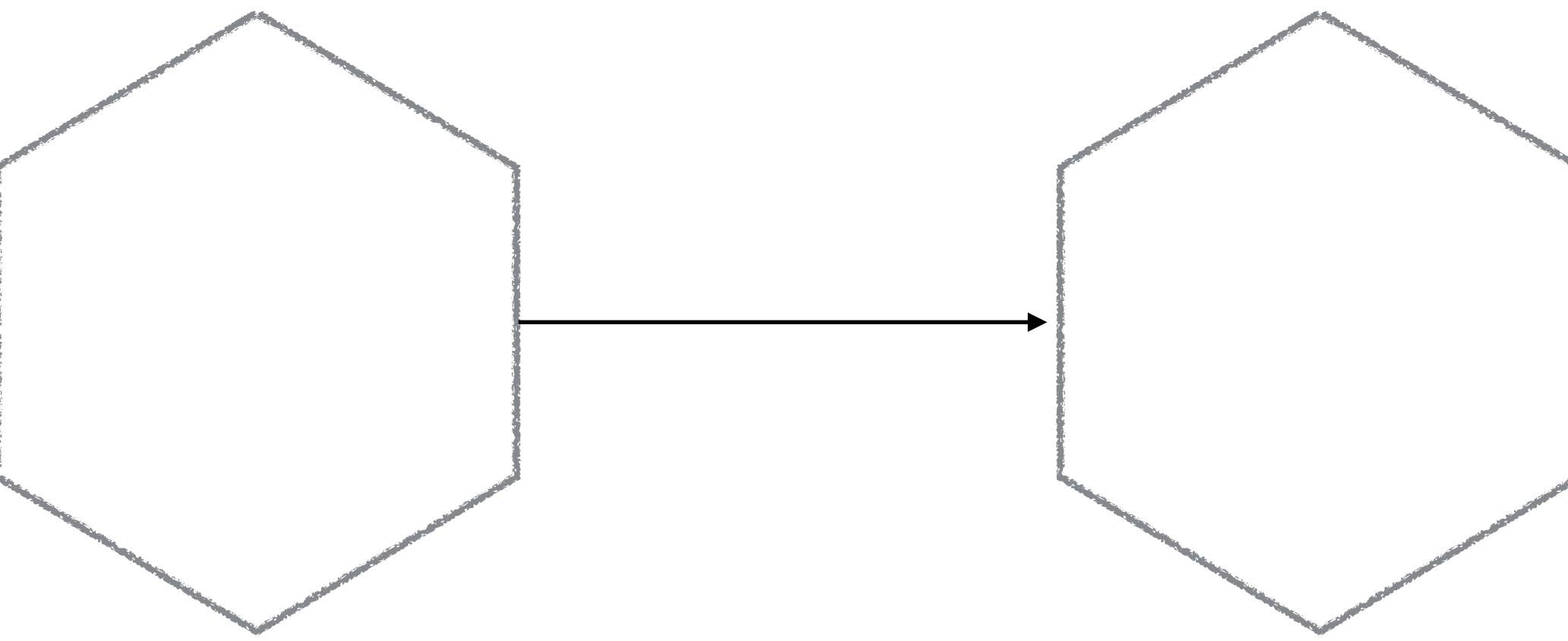
**Istio**

**<https://istio.io>**

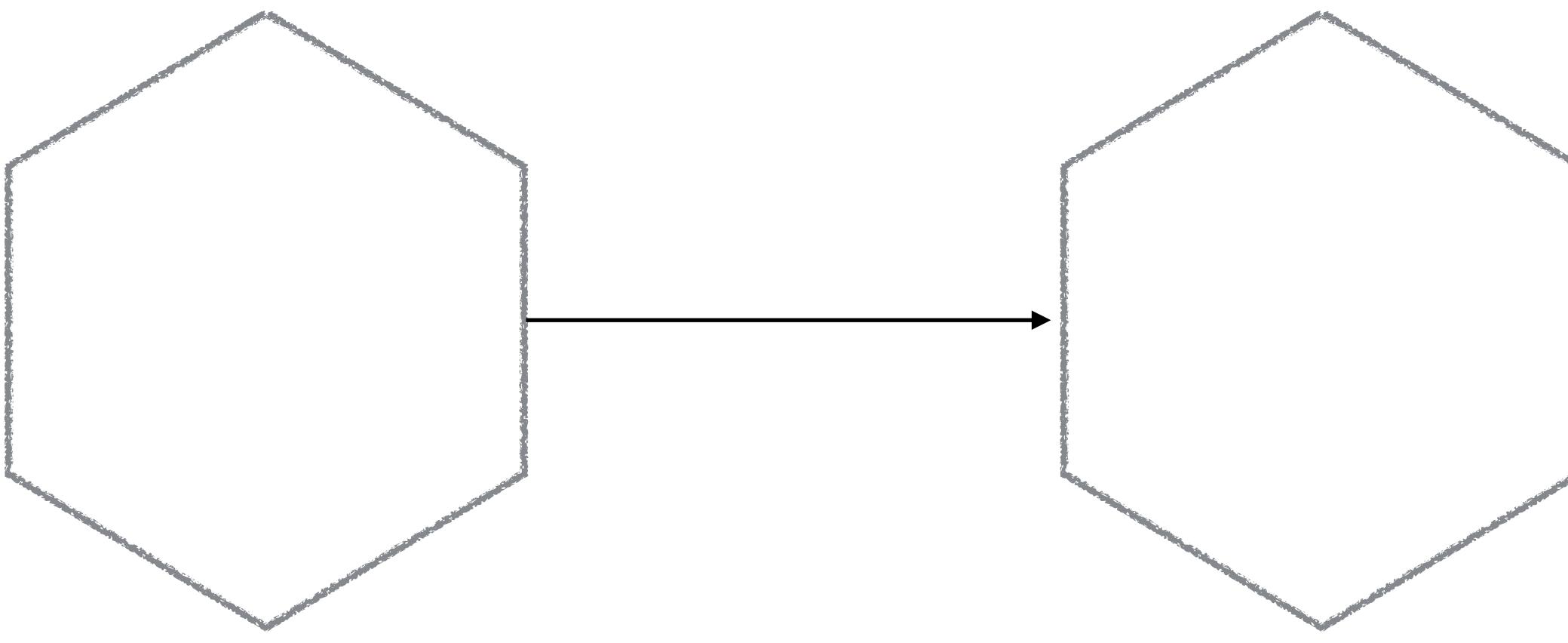
# COMMON CONNECTION CONCERNS



# COMMON CONNECTION CONCERNS

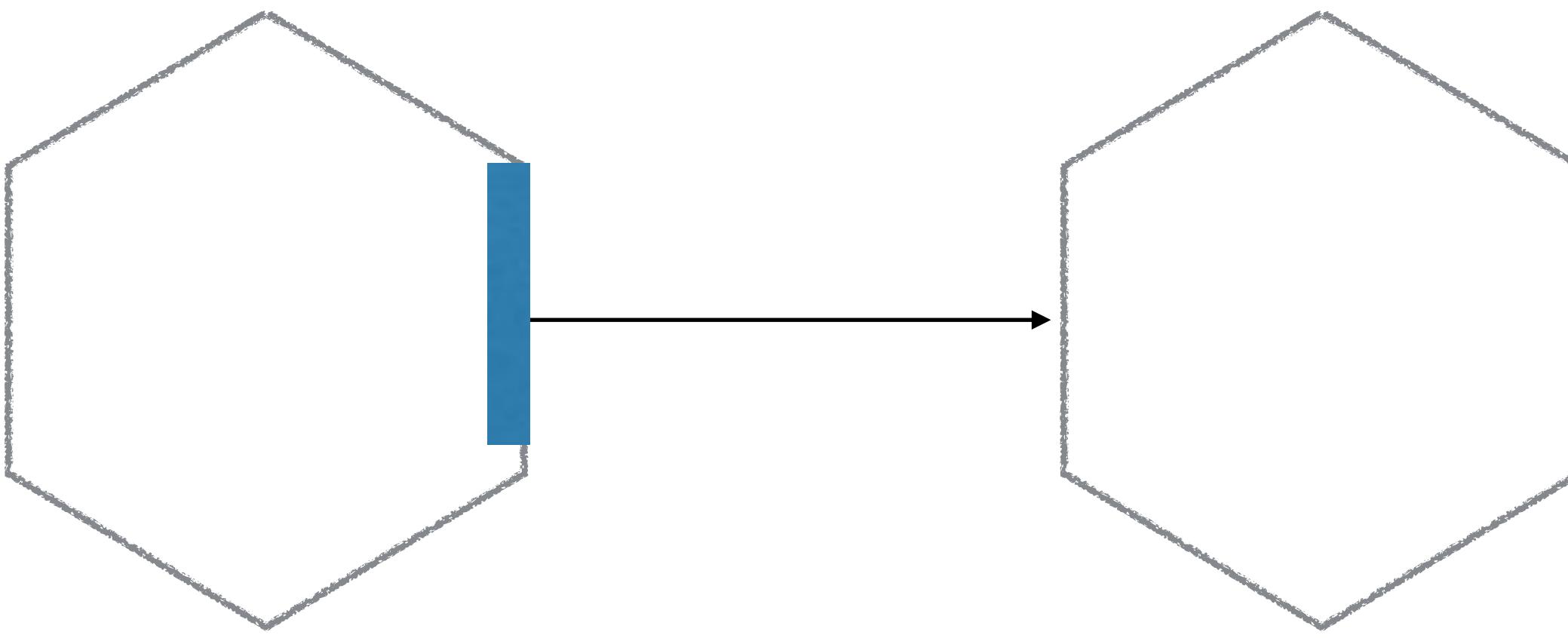


# COMMON CONNECTION CONCERNS



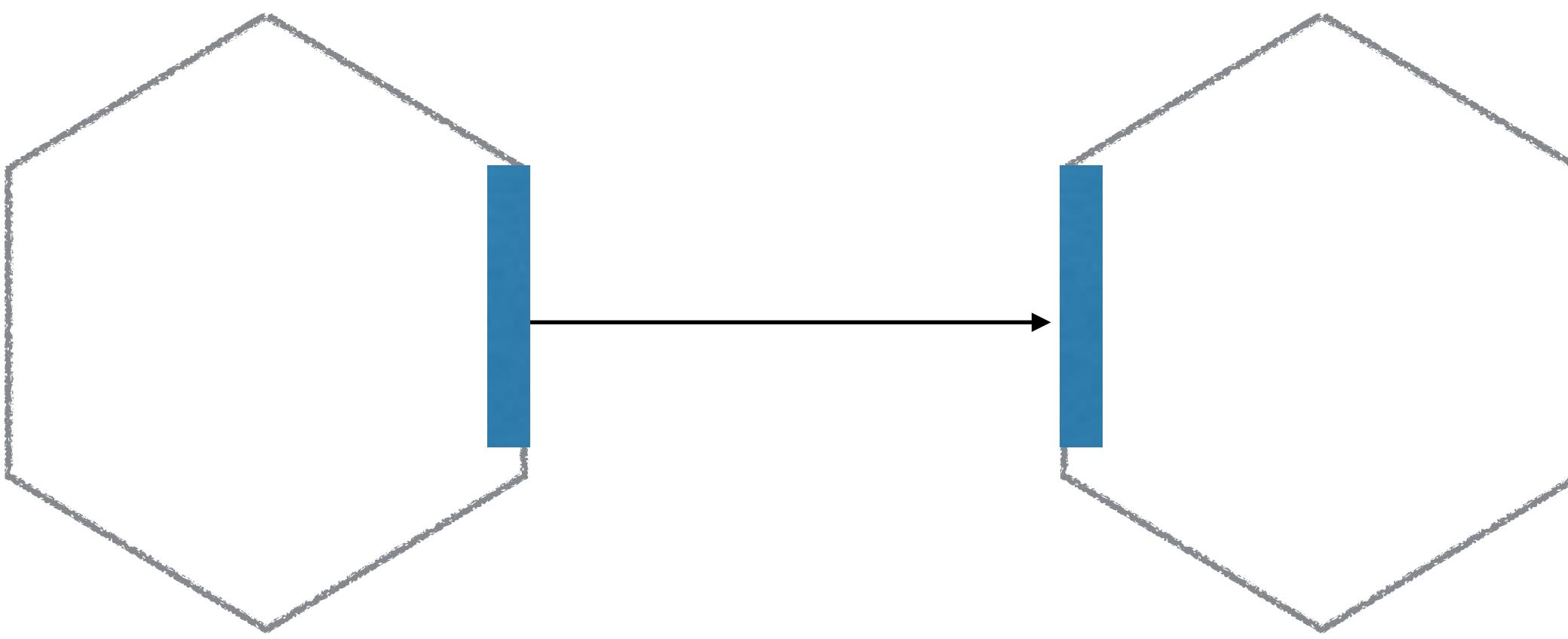
Tracing

# COMMON CONNECTION CONCERNS



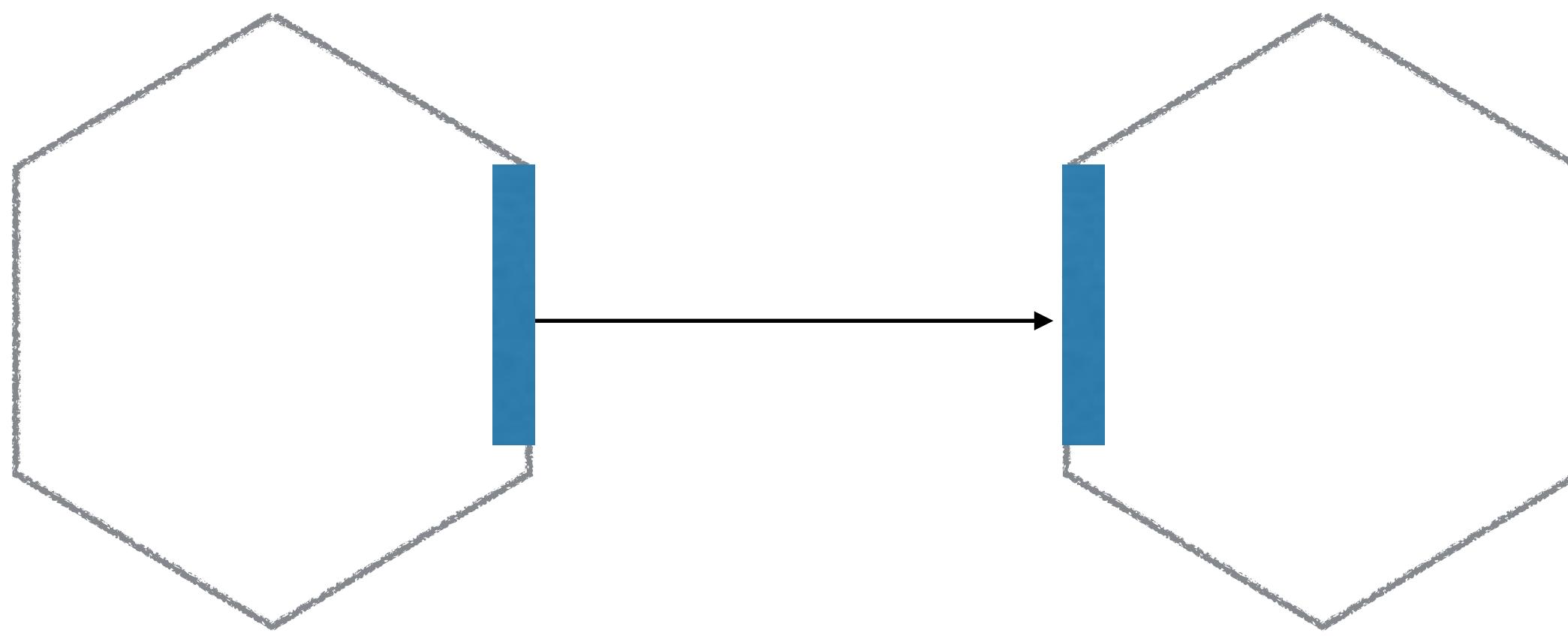
Tracing

# COMMON CONNECTION CONCERNS



Tracing

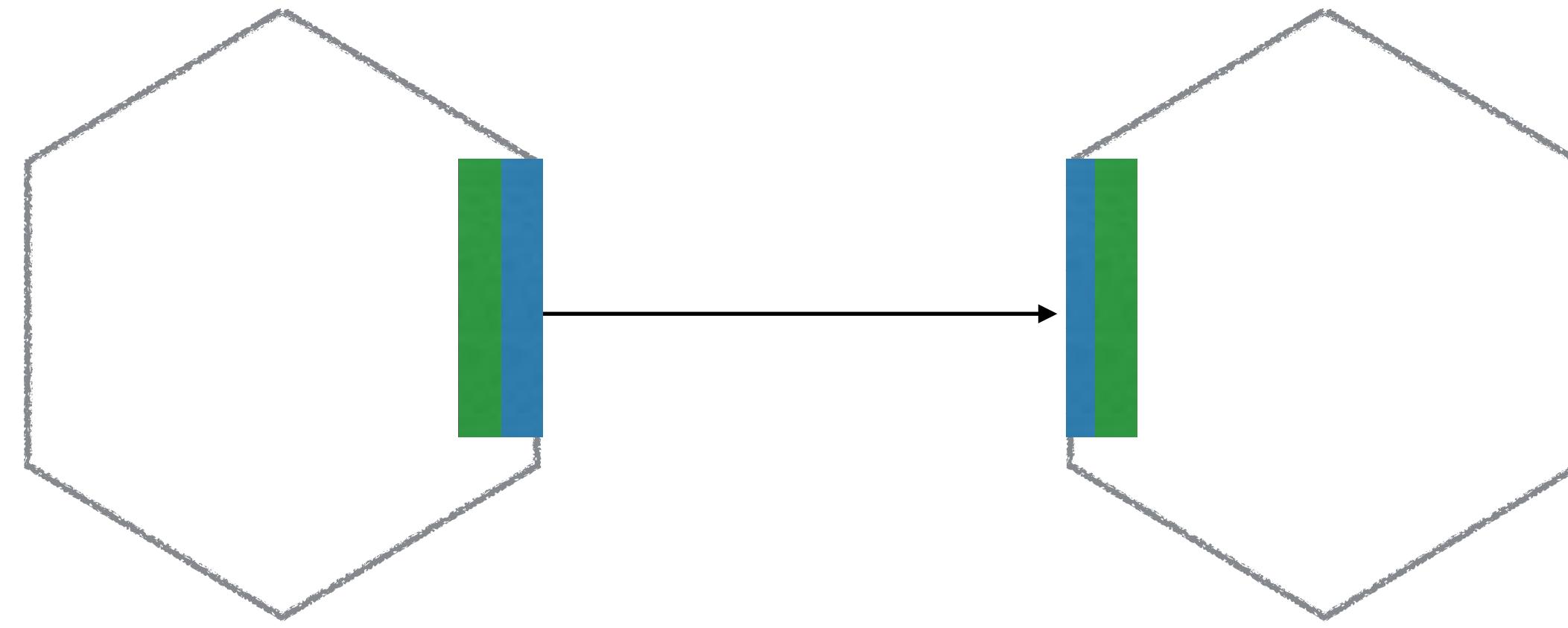
# COMMON CONNECTION CONCERNS



Tracing

Load Balancing & Service Discovery

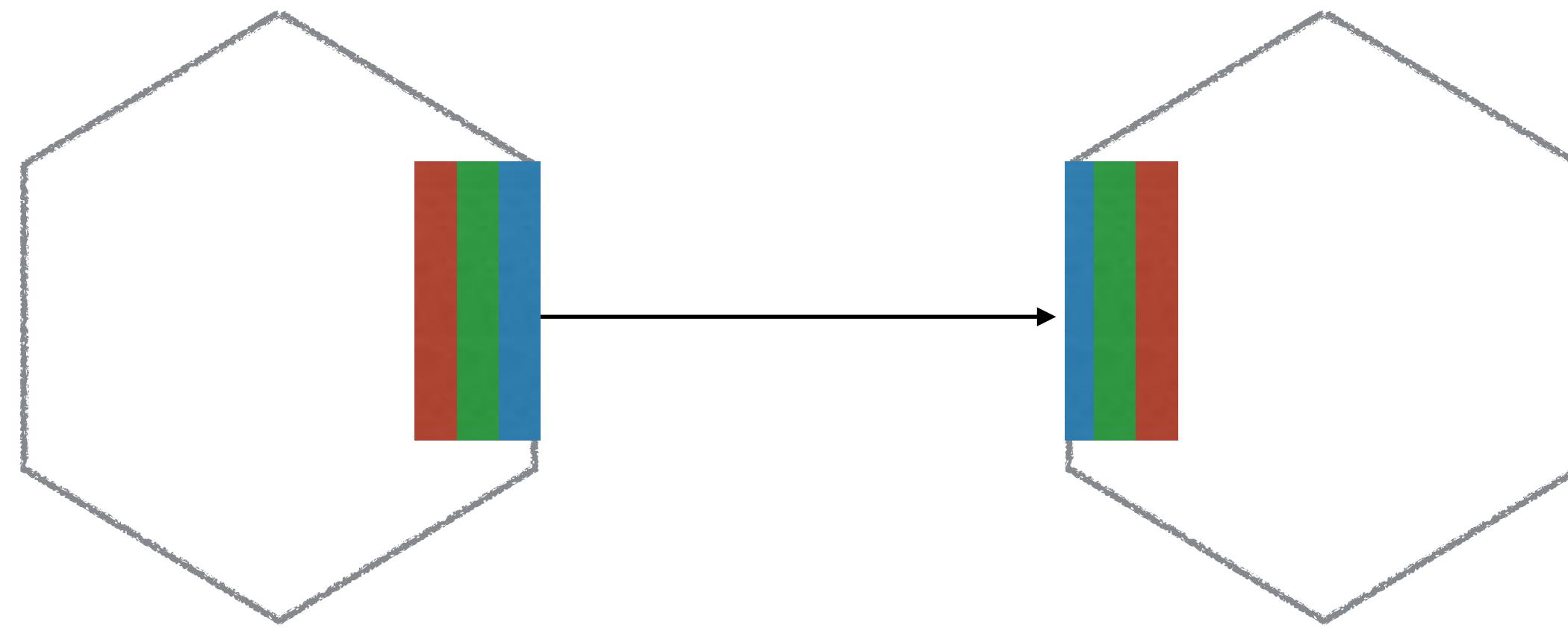
# COMMON CONNECTION CONCERNS



Tracing

Load Balancing & Service Discovery

# COMMON CONNECTION CONCERNS

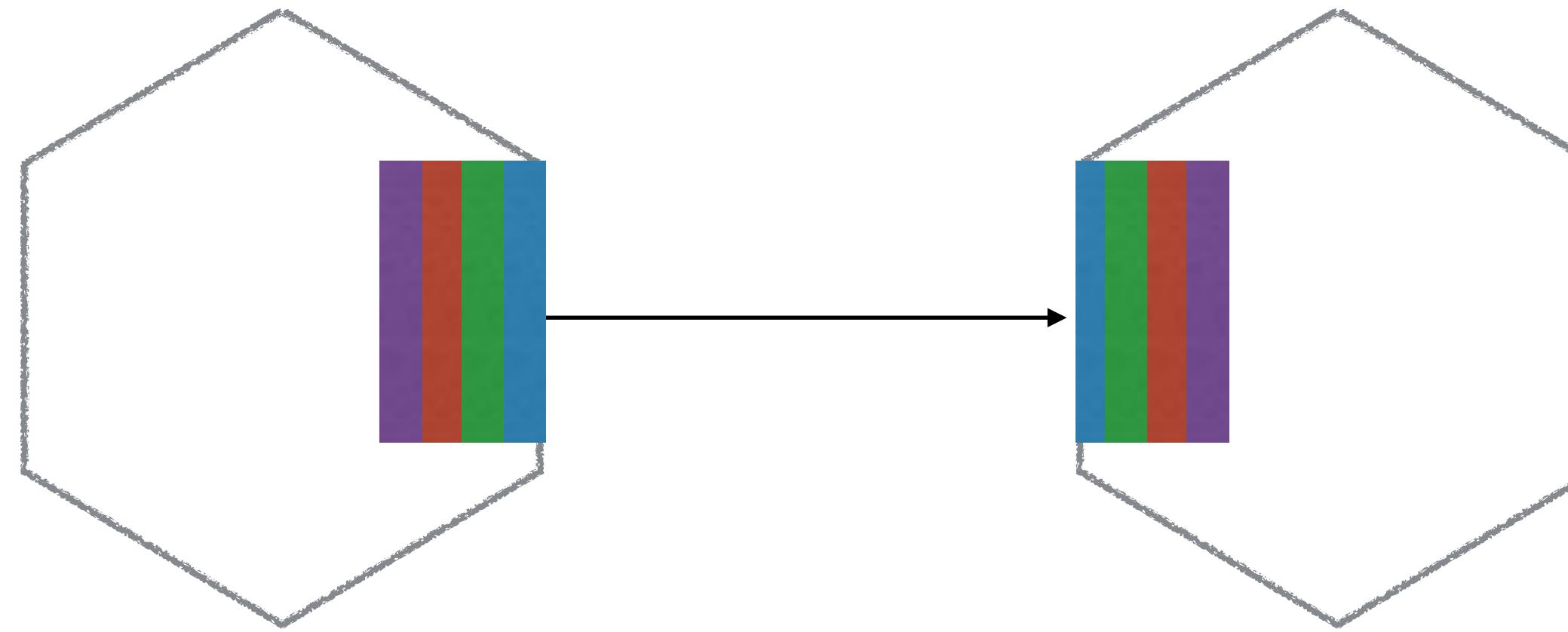


Tracing

Load Balancing & Service Discovery

Authorisation & Authentication

# COMMON CONNECTION CONCERNS



Tracing

Load Balancing & Service Discovery

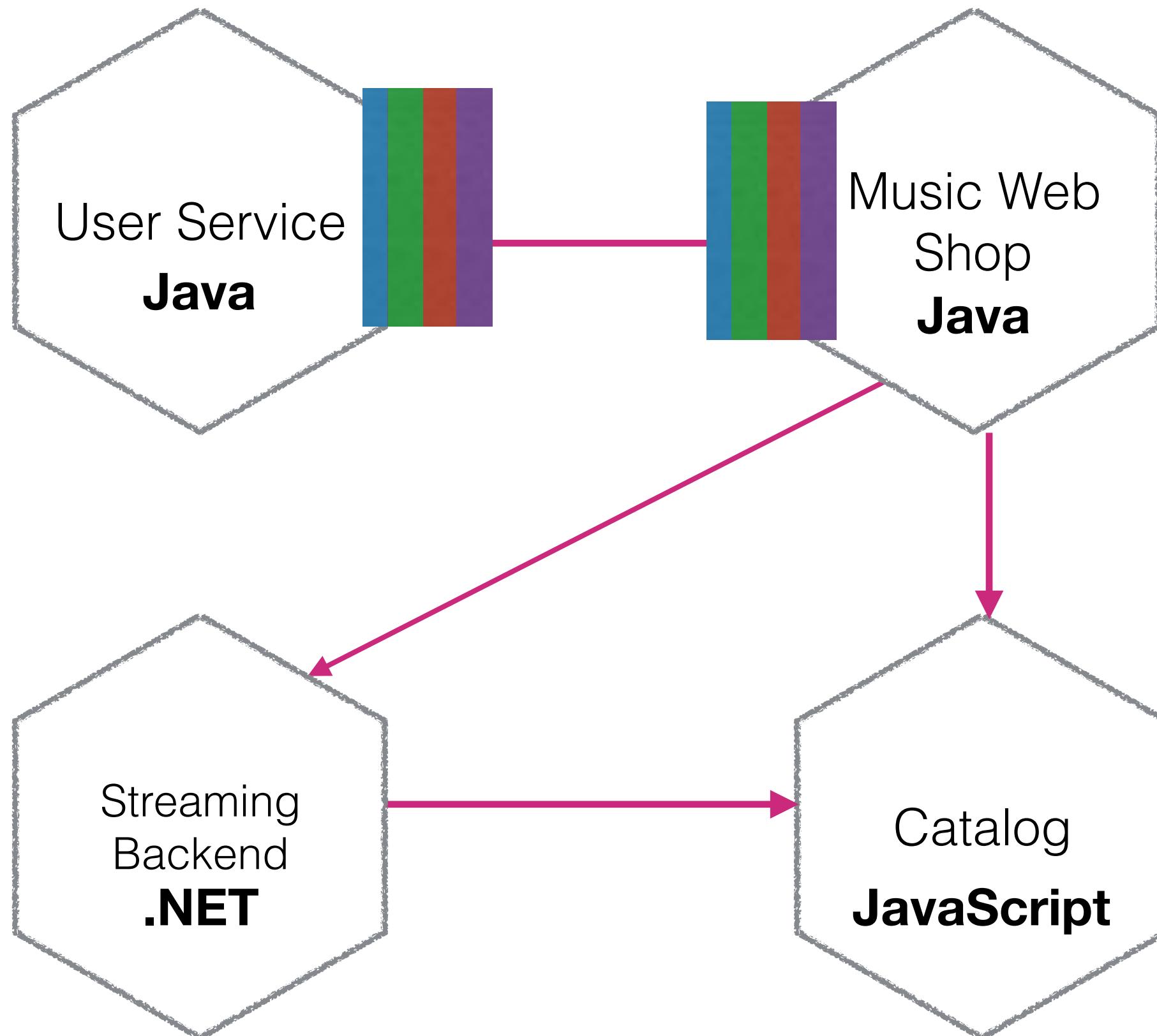
Authorisation & Authentication

Connection Resilience & Retry

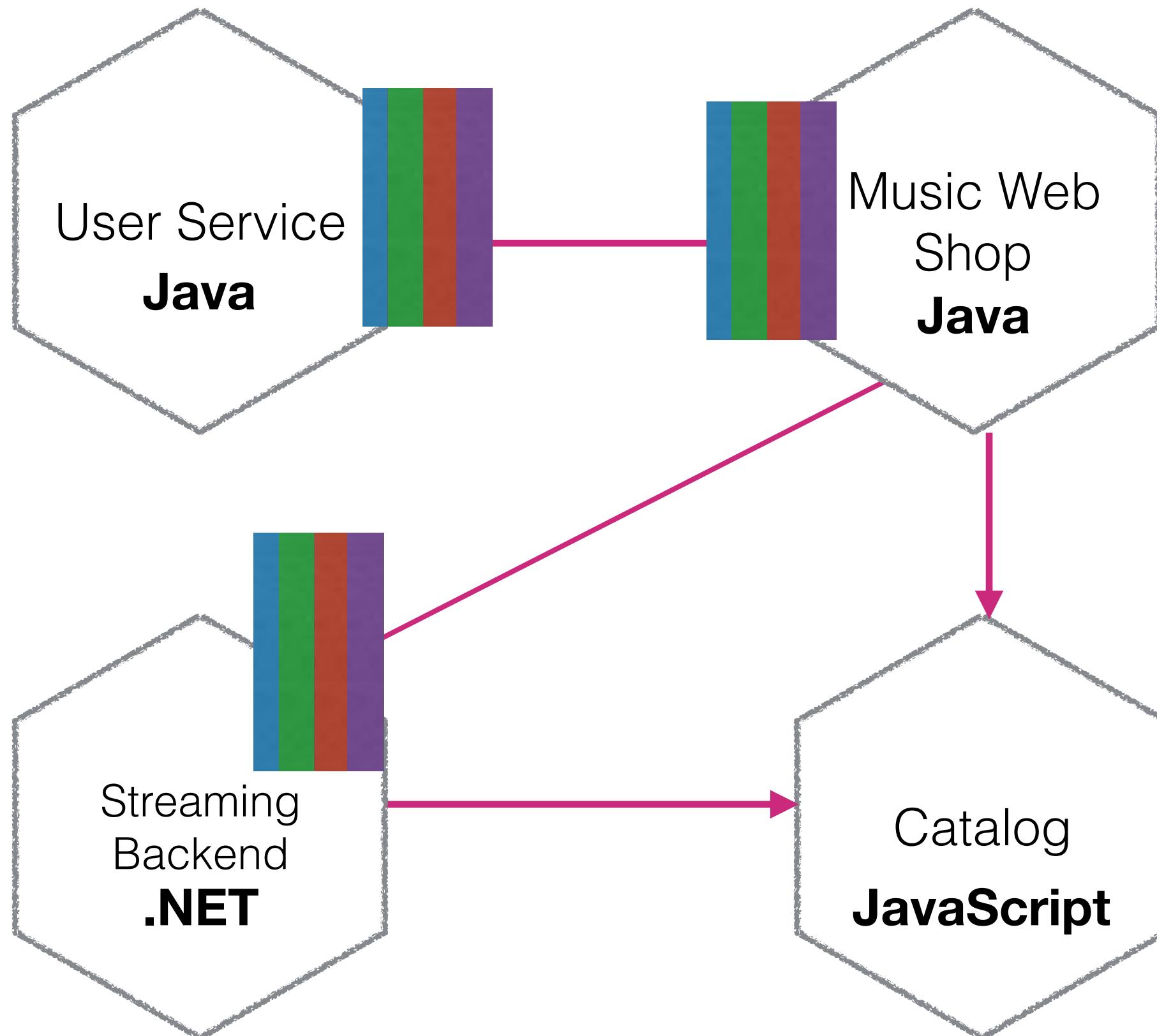
## COMMON MICROSERVICE FRAMEWORKS



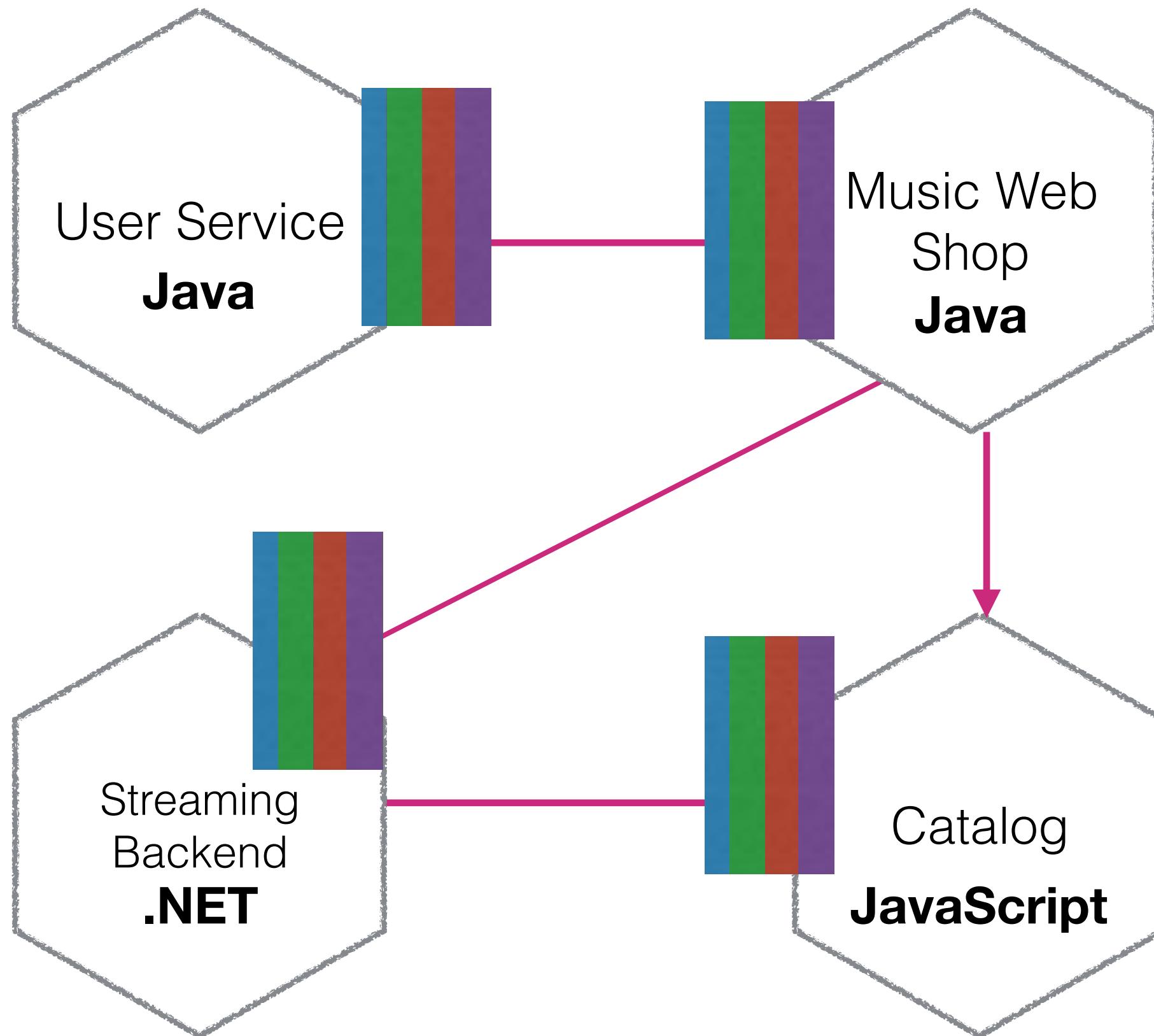
# POLYGLOT?



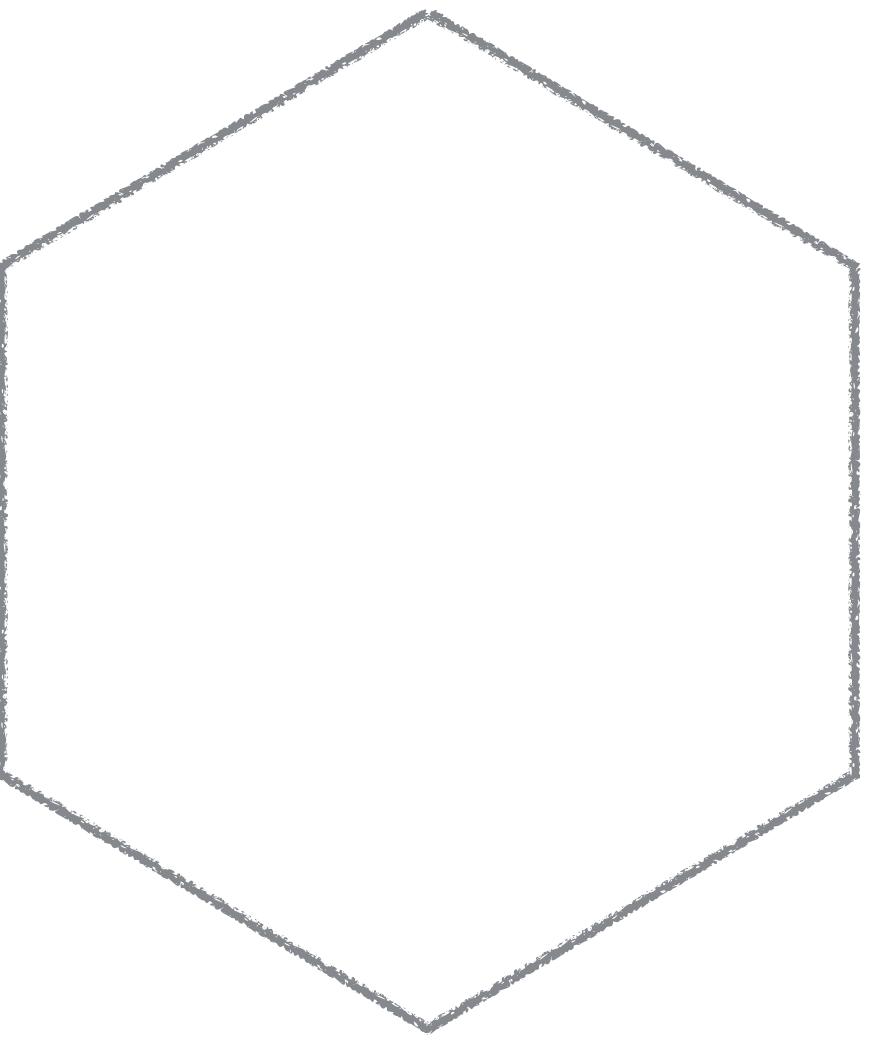
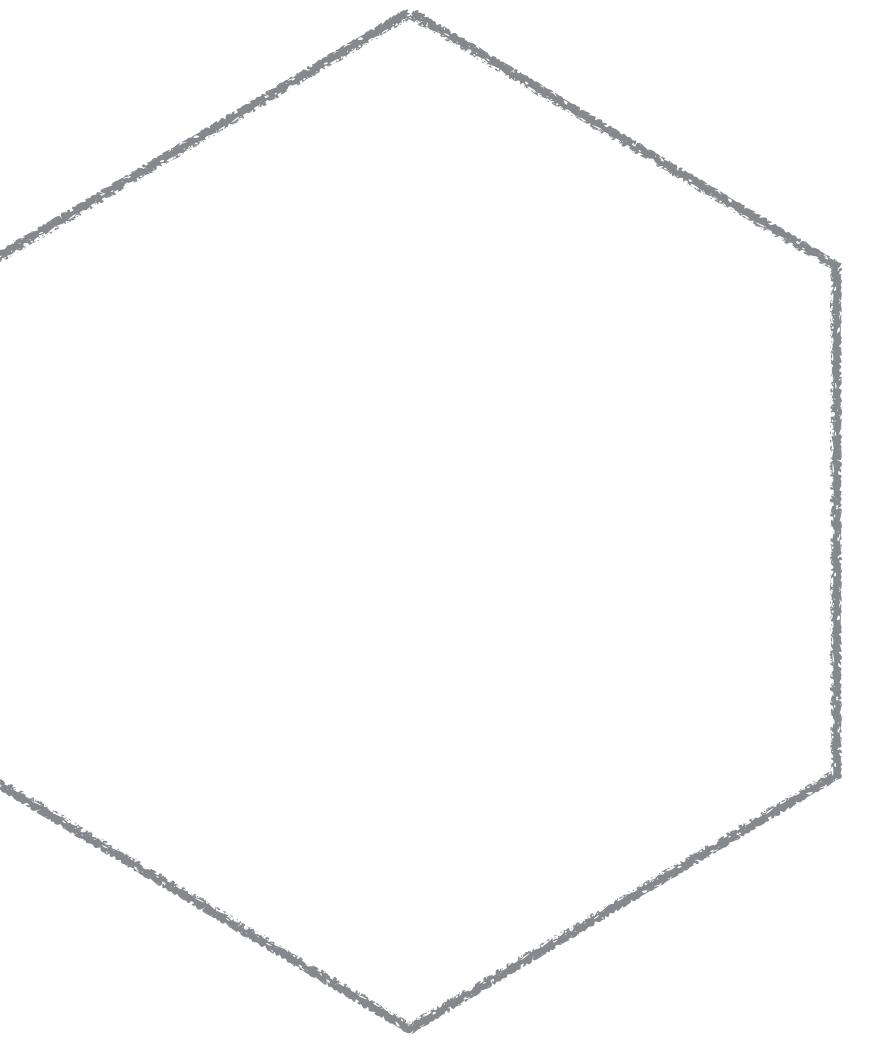
# POLYGLOT?



# POLYGLOT?



# VERSION DRIFT



# VERSION DRIFT



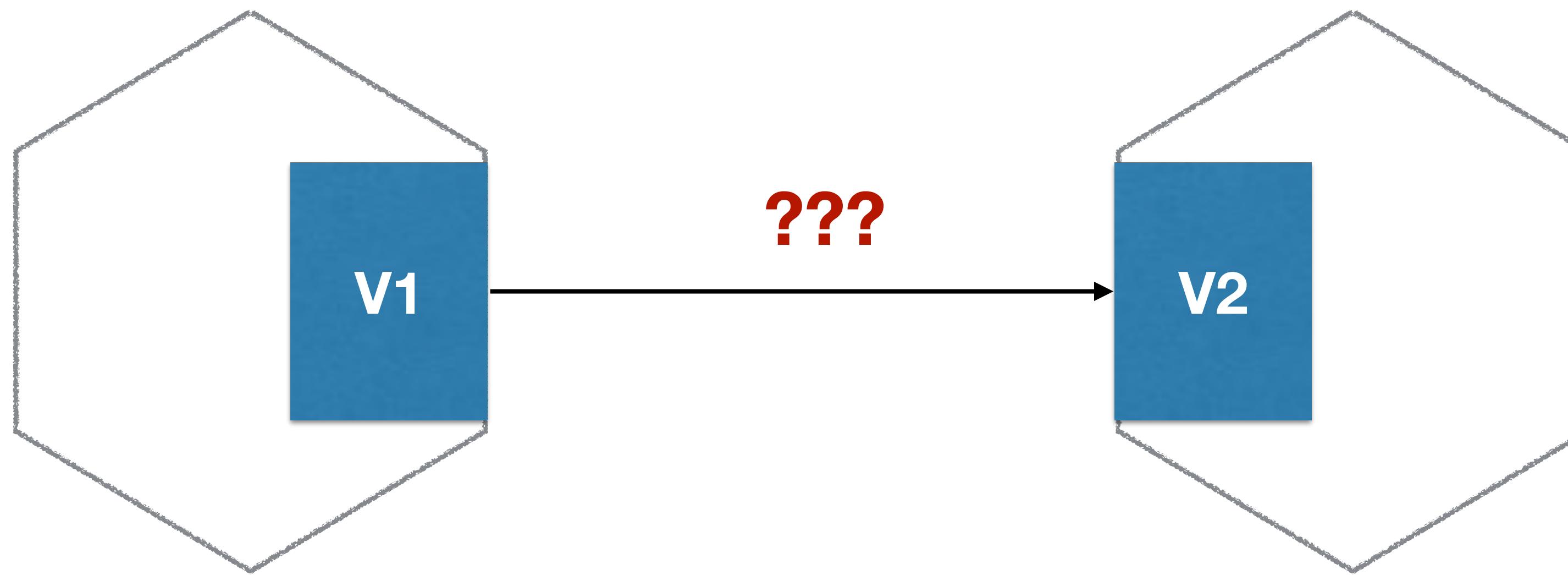
# VERSION DRIFT



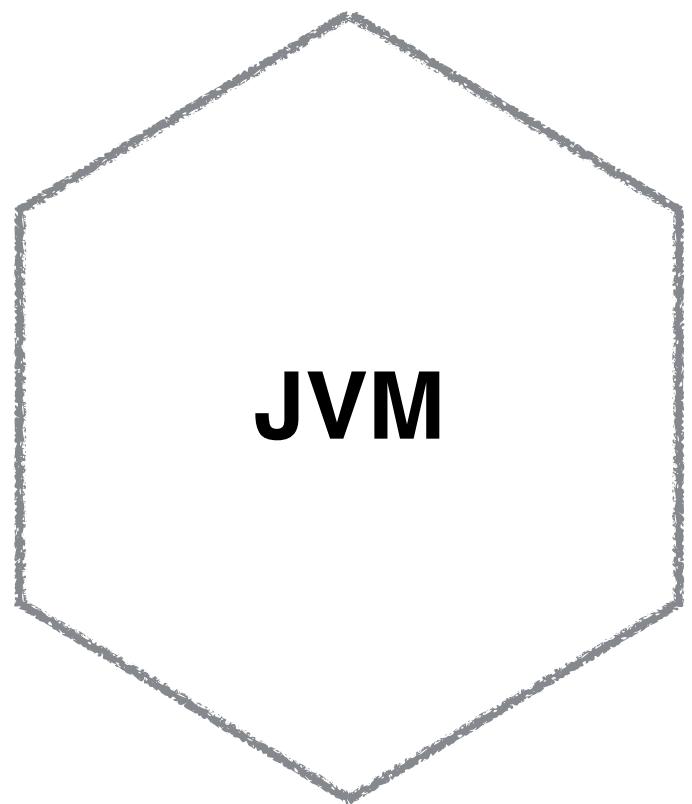
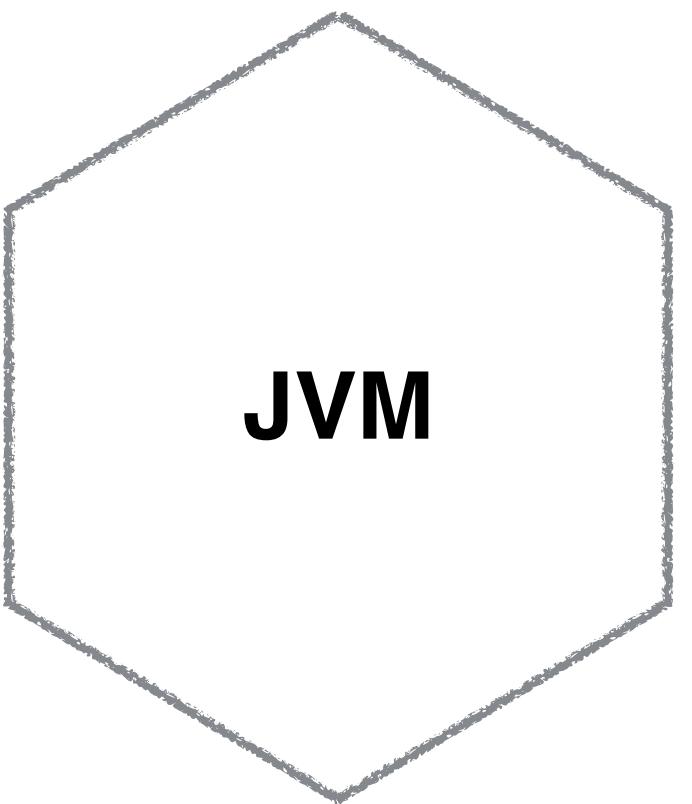
## VERSION DRIFT



## VERSION DRIFT

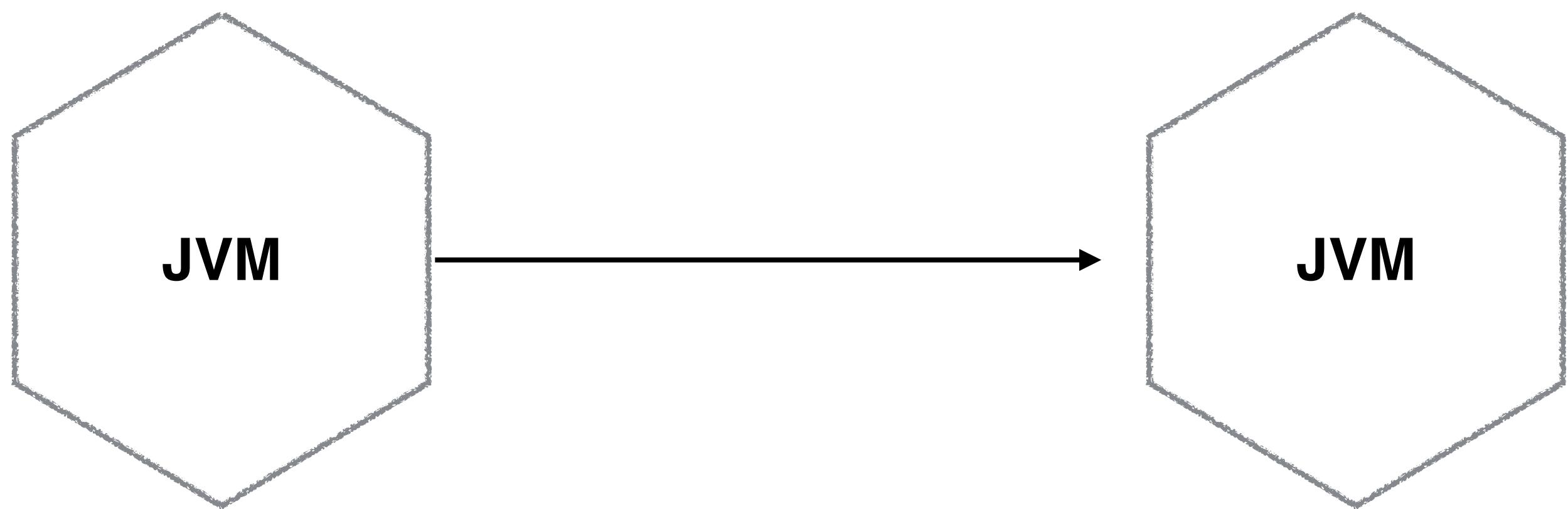


## NETFLIX - ENFORCEMENT OF REUSE



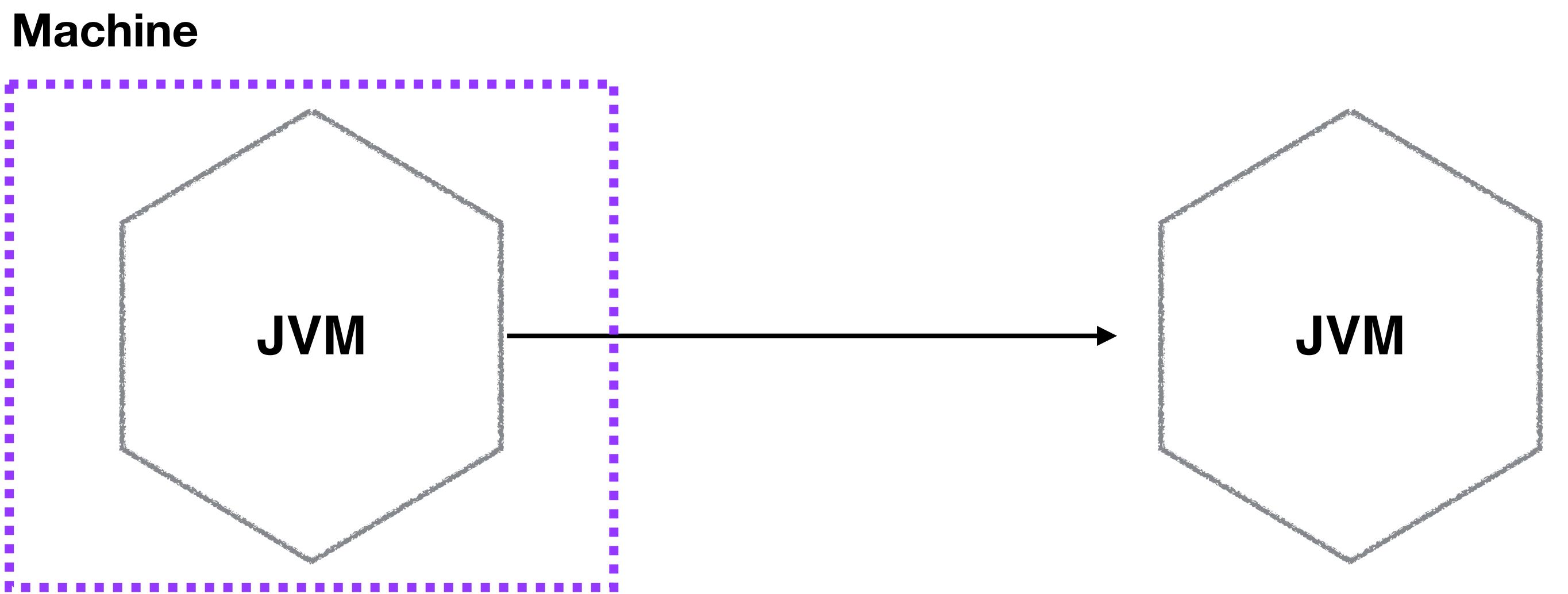
NETFLIX

## NETFLIX - ENFORCEMENT OF REUSE



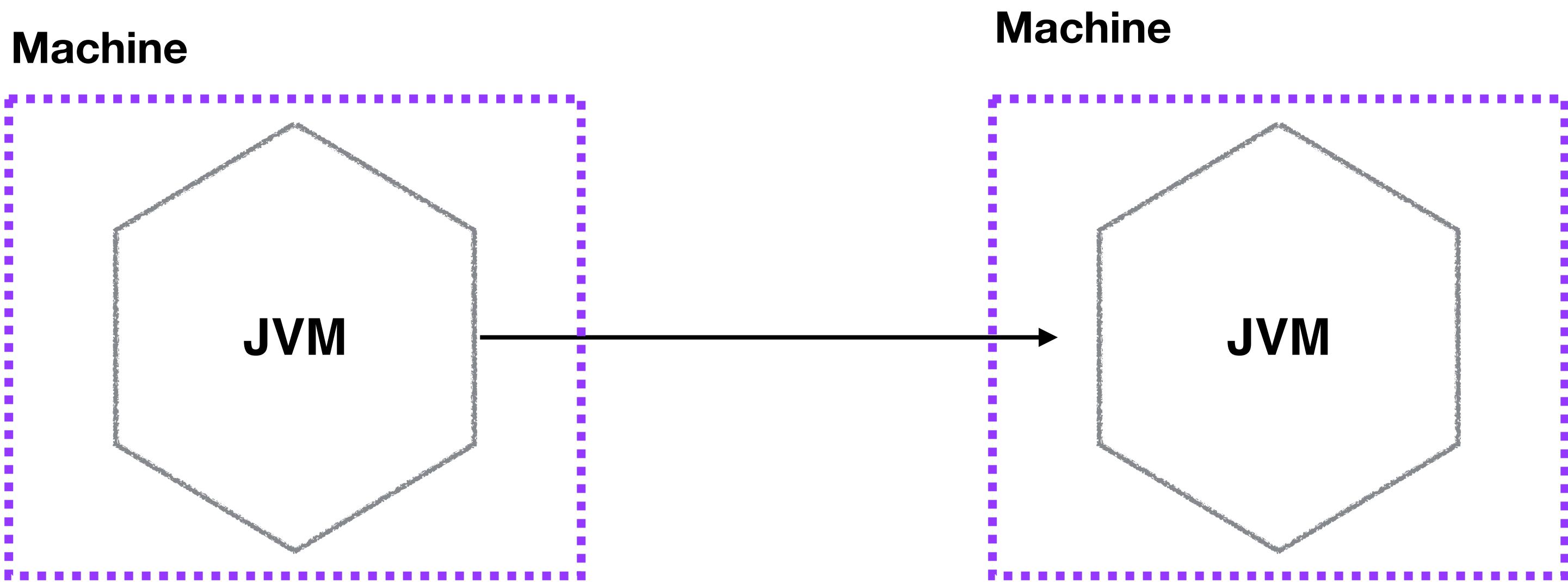
NETFLIX

## NETFLIX - ENFORCEMENT OF REUSE



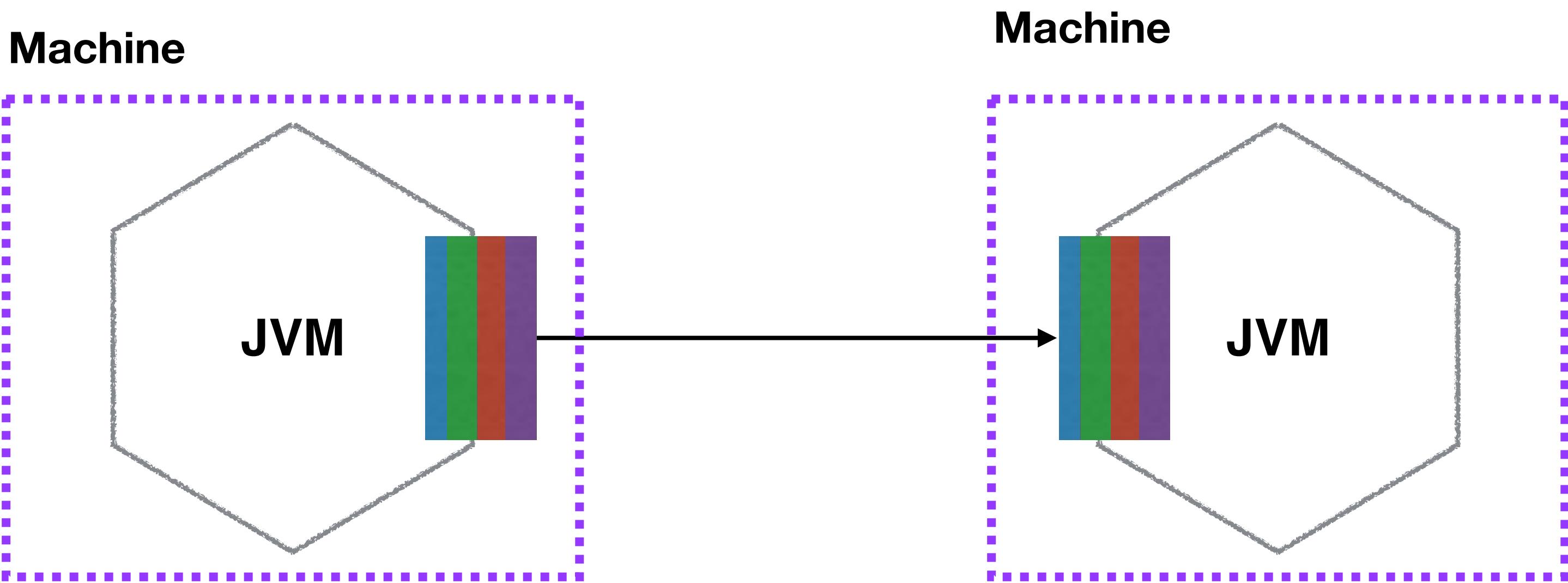
**NETFLIX**

## NETFLIX - ENFORCEMENT OF REUSE



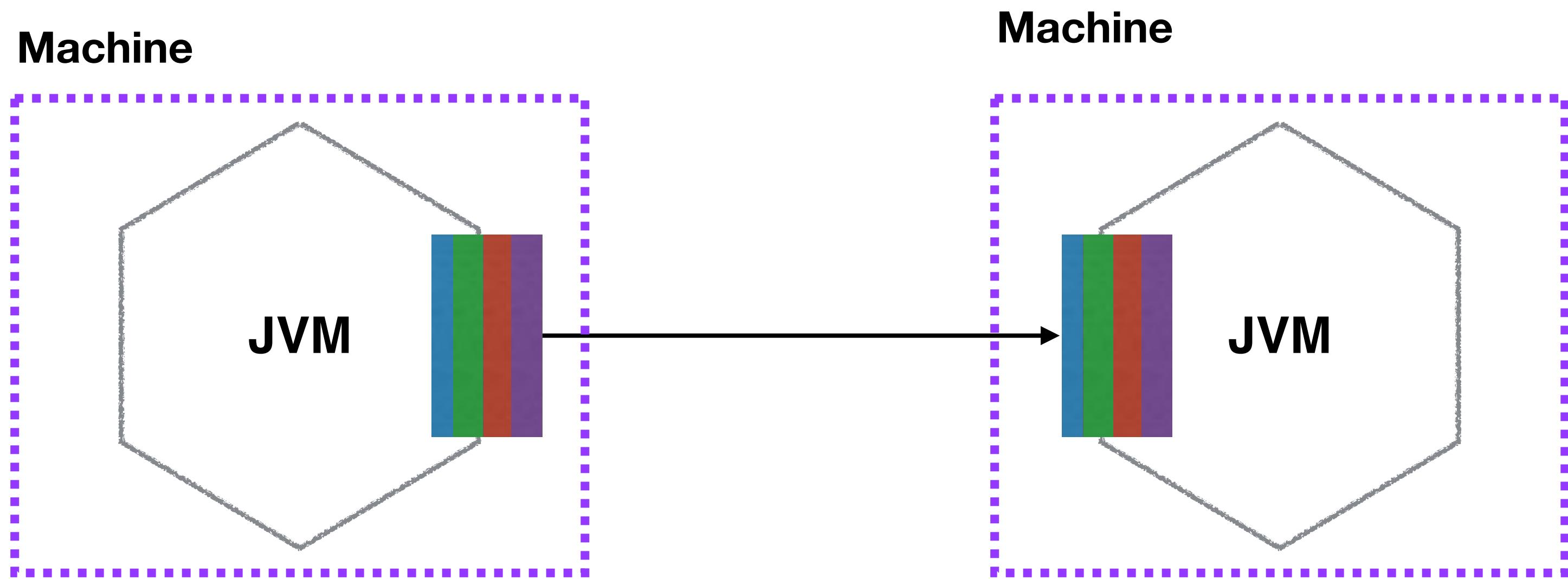
NETFLIX

## NETFLIX - ENFORCEMENT OF REUSE



NETFLIX

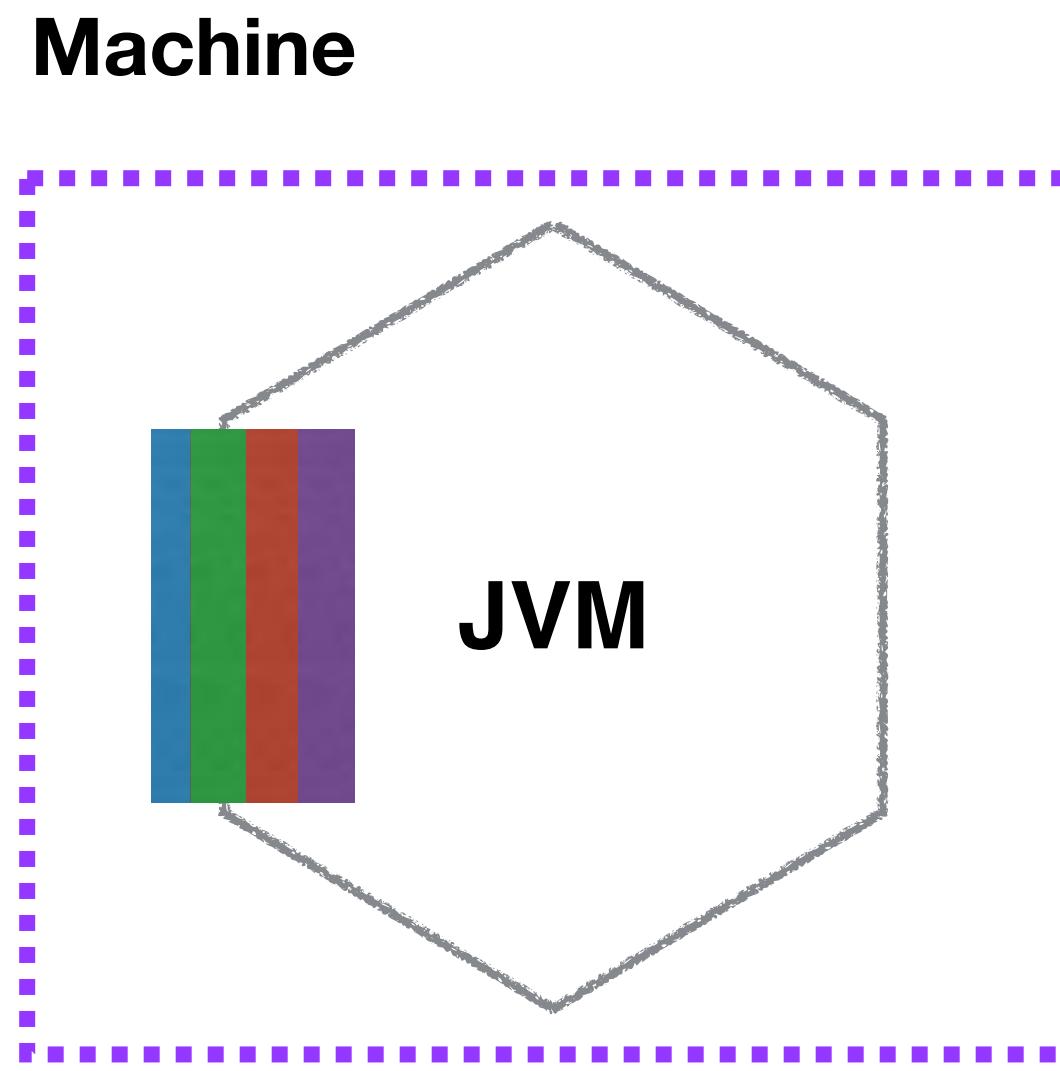
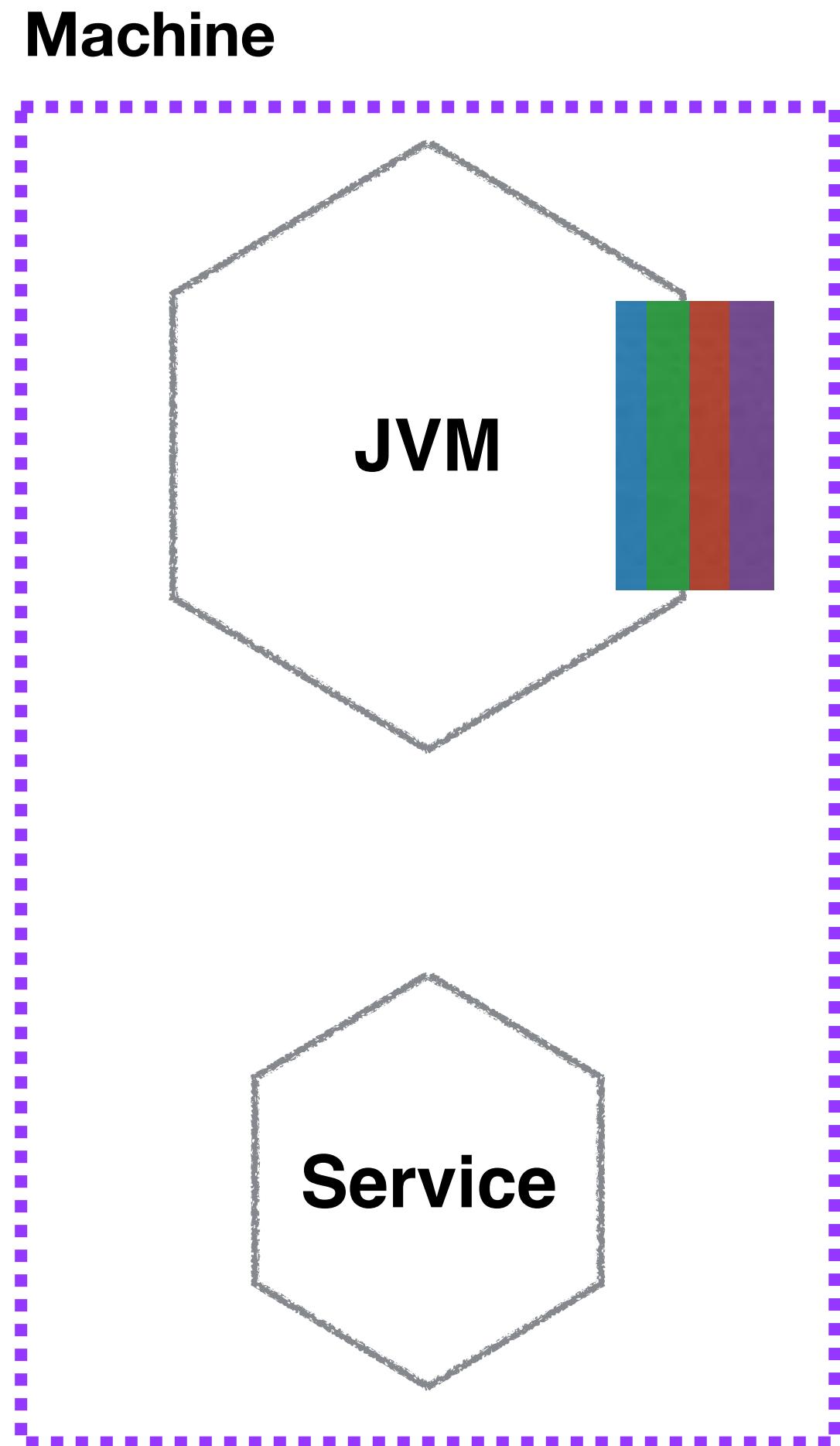
## NETFLIX - ENFORCEMENT OF REUSE



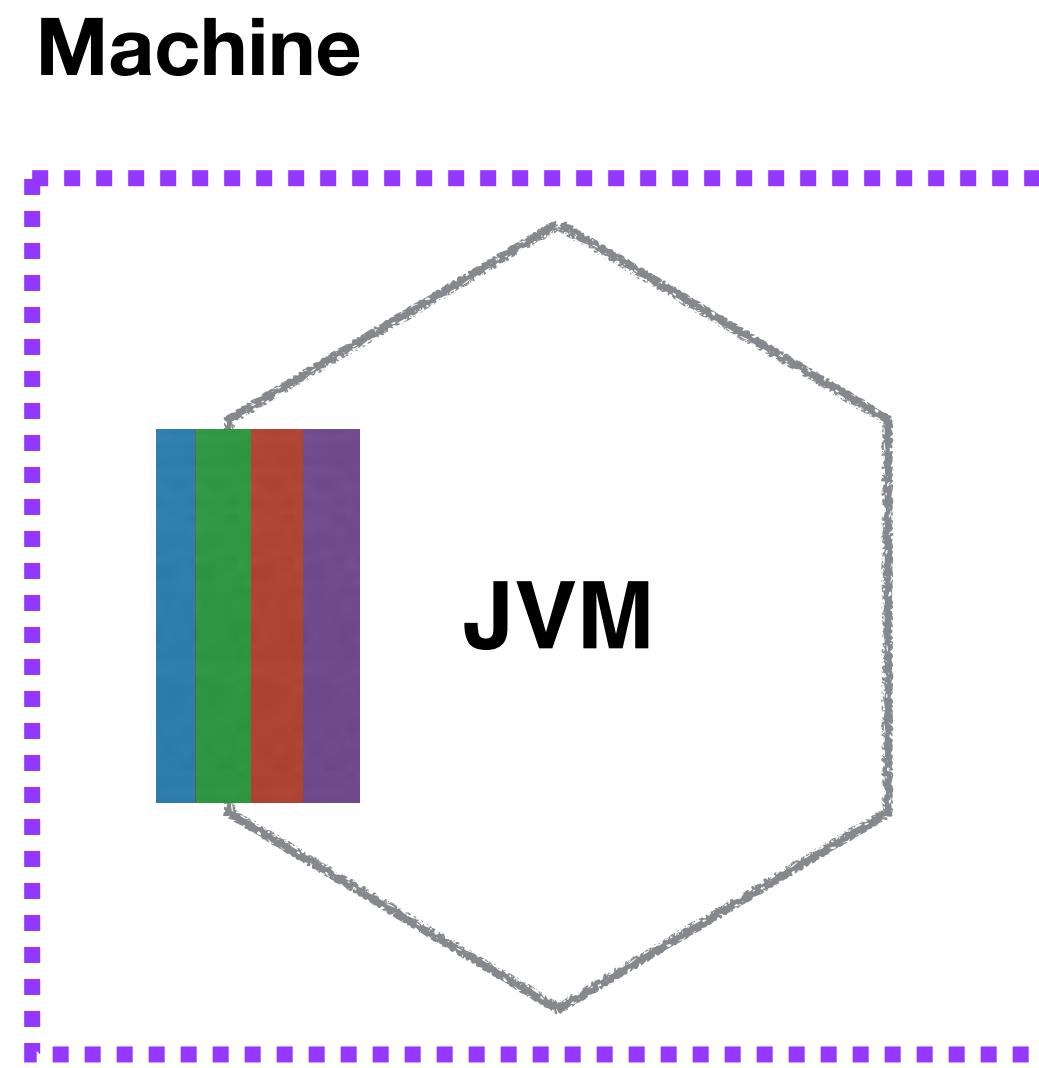
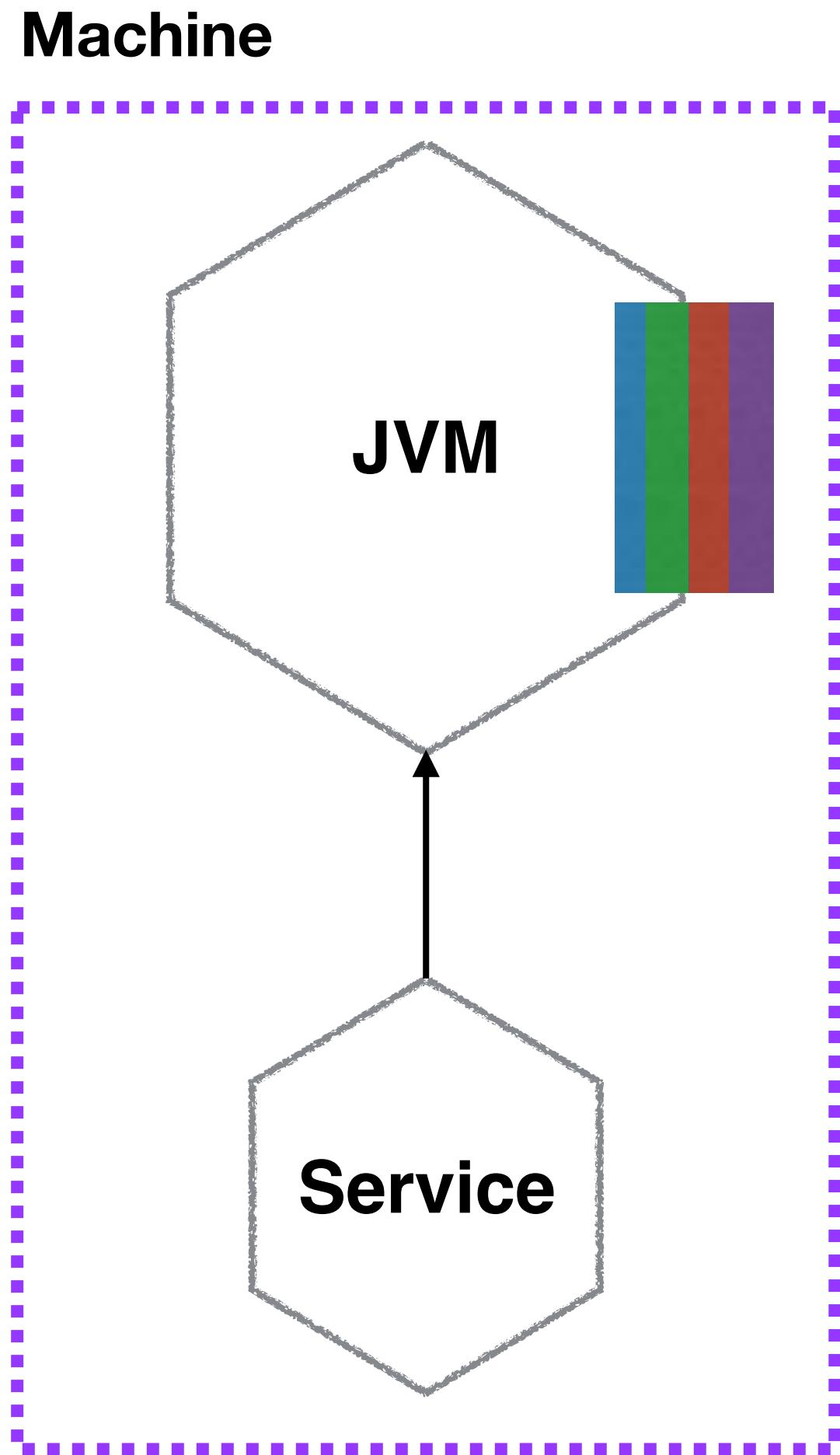
NETFLIX

What about non-JVM  
languages?

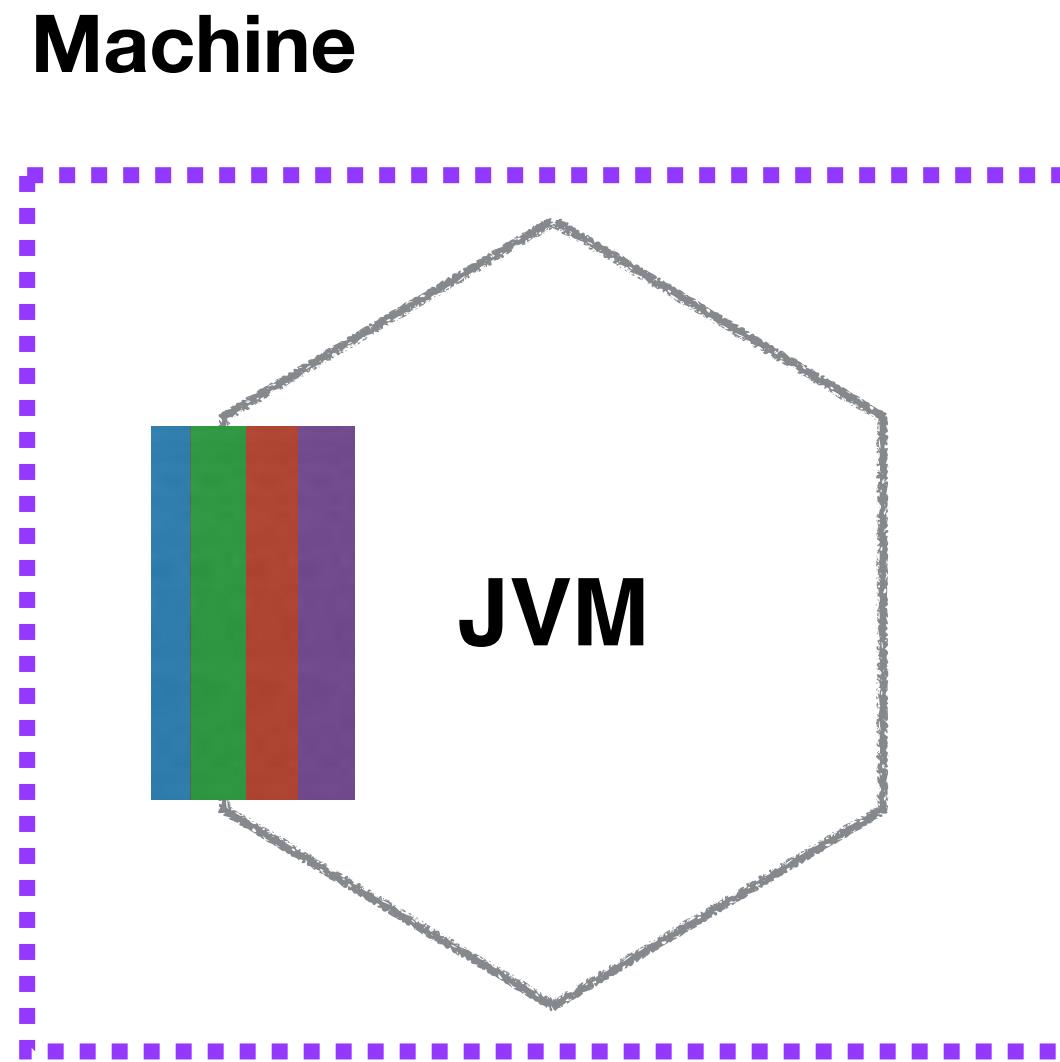
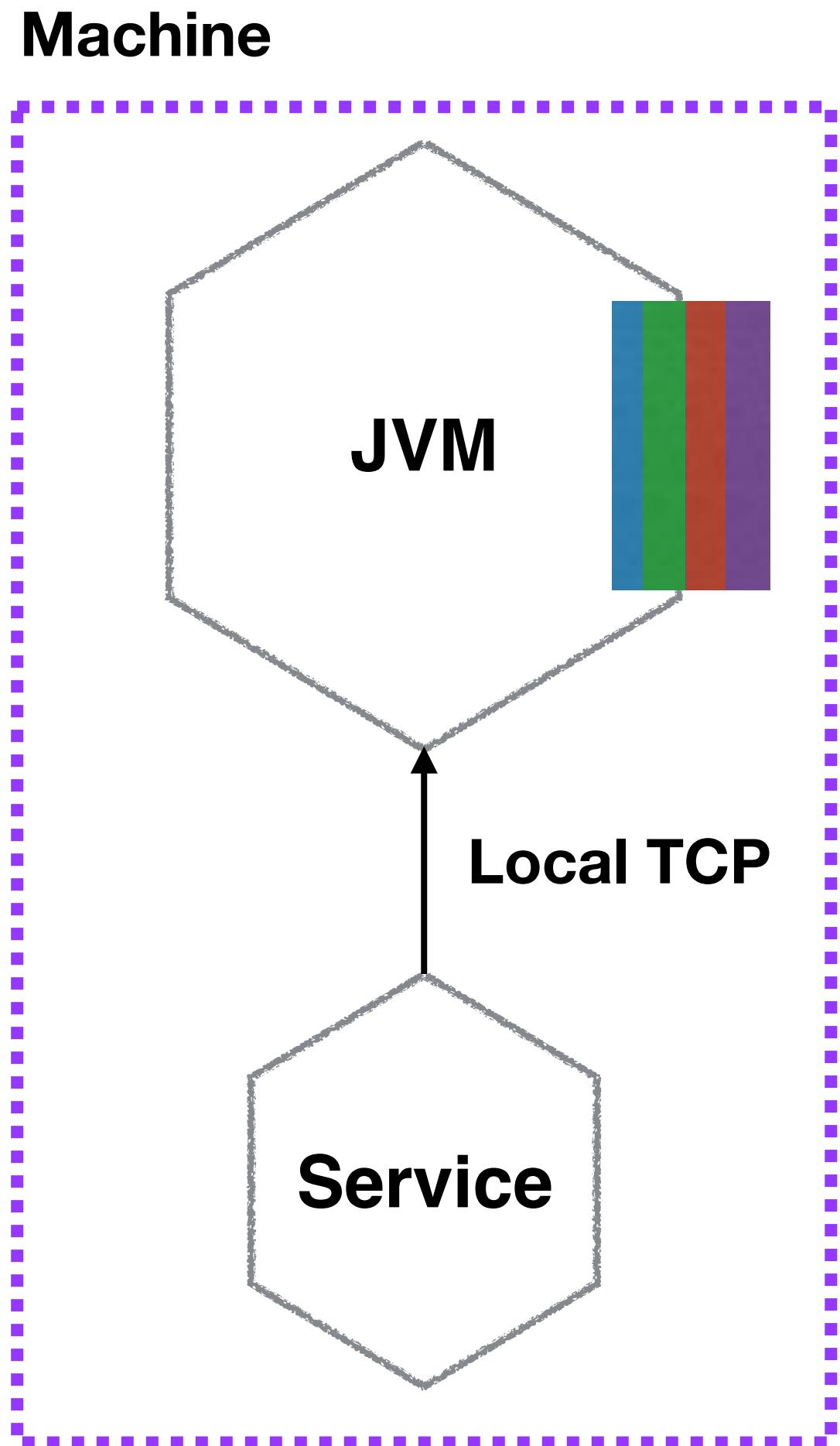
# NETFLIX - SIDECAR PATTERN



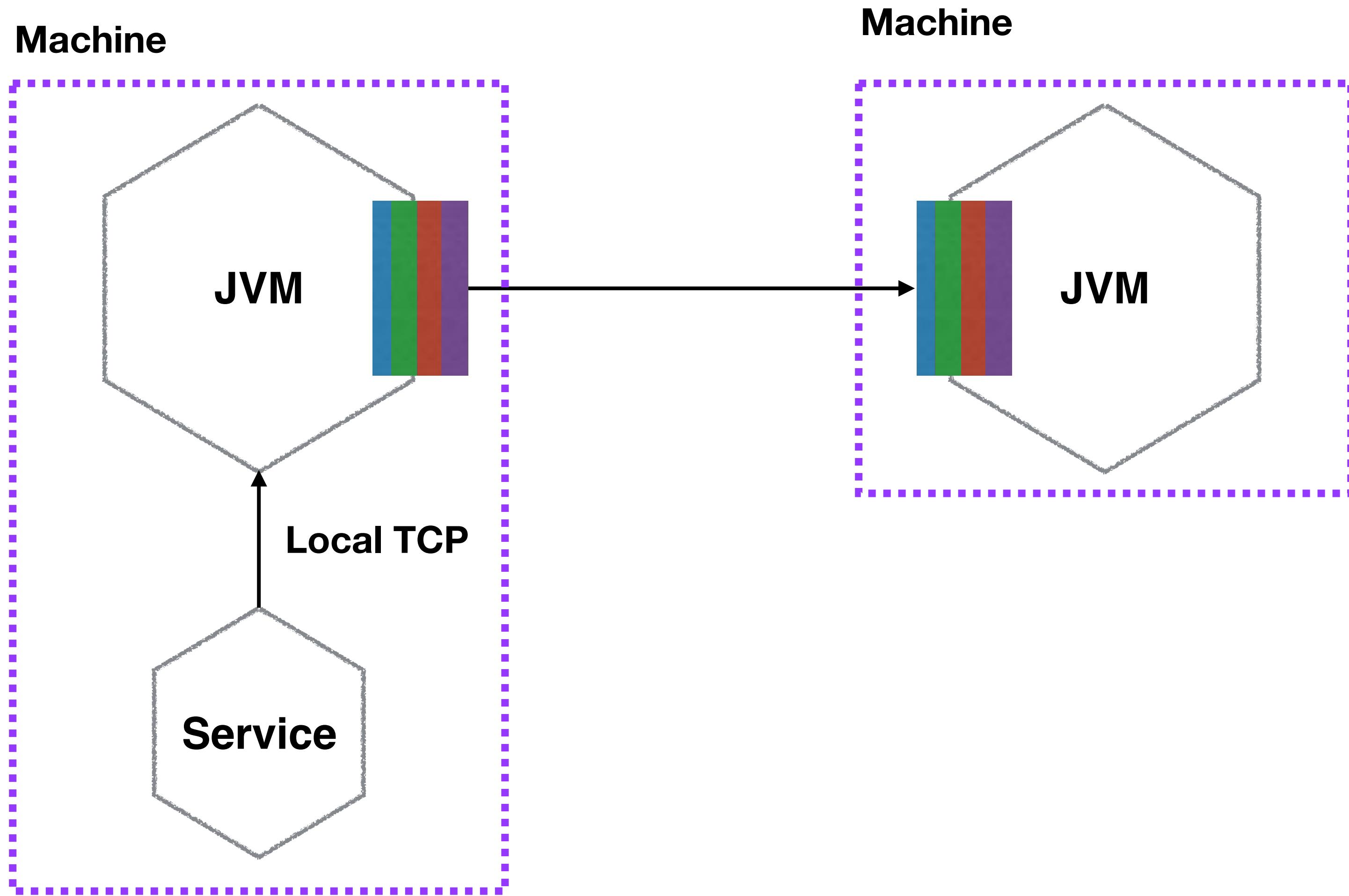
# NETFLIX - SIDECAR PATTERN



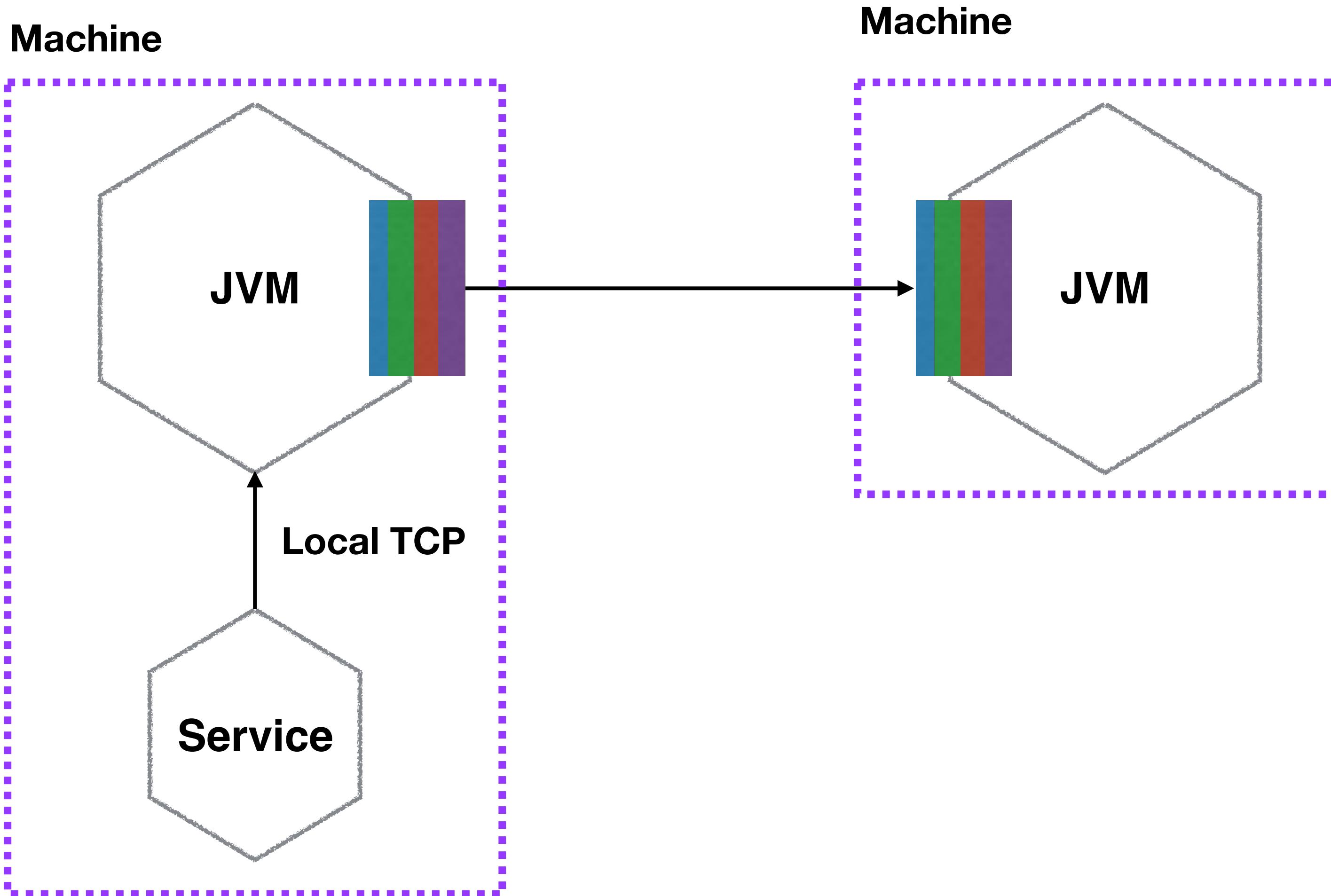
# NETFLIX - SIDECAR PATTERN



# NETFLIX - SIDECAR PATTERN

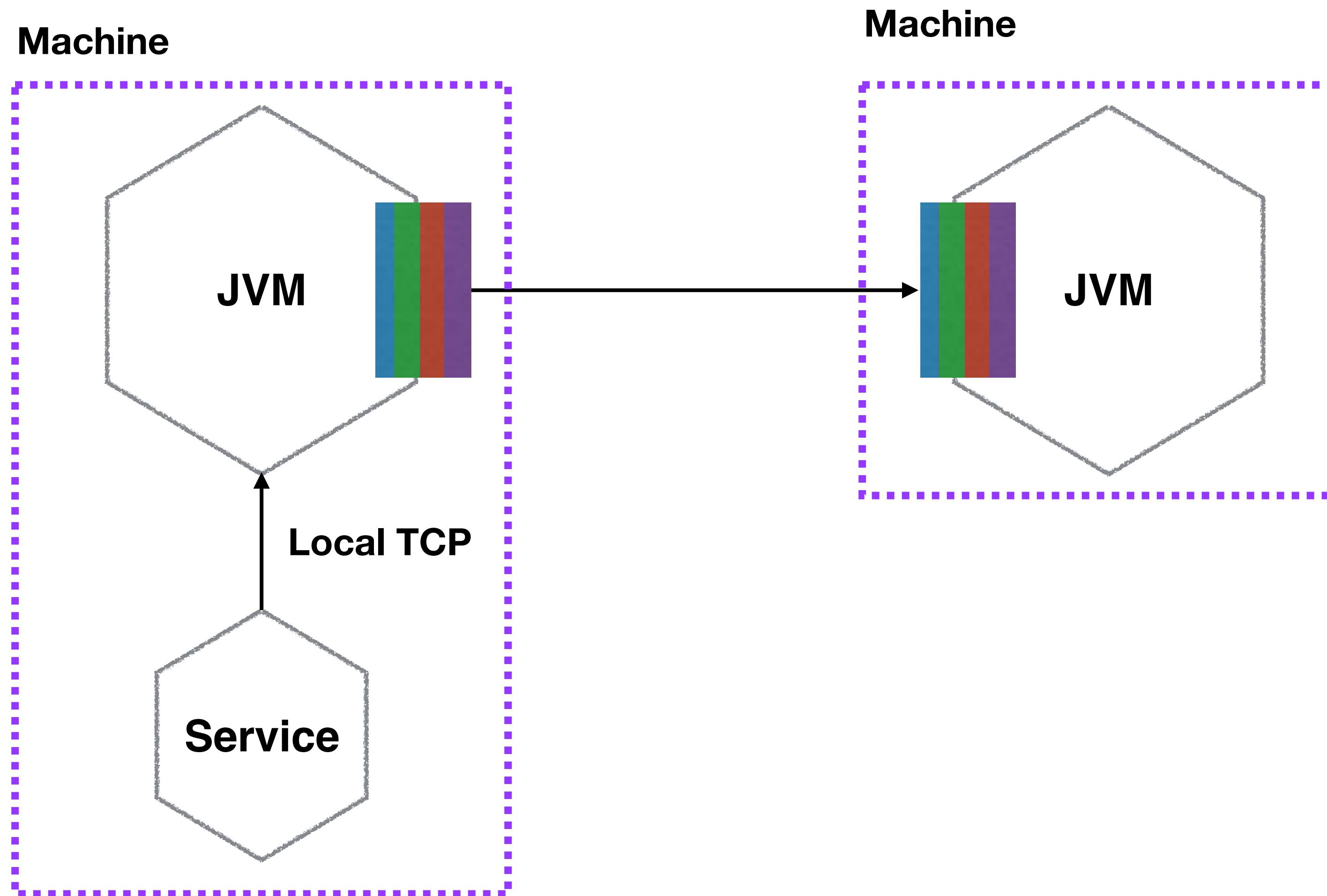


## NETFLIX - SIDECAR PATTERN



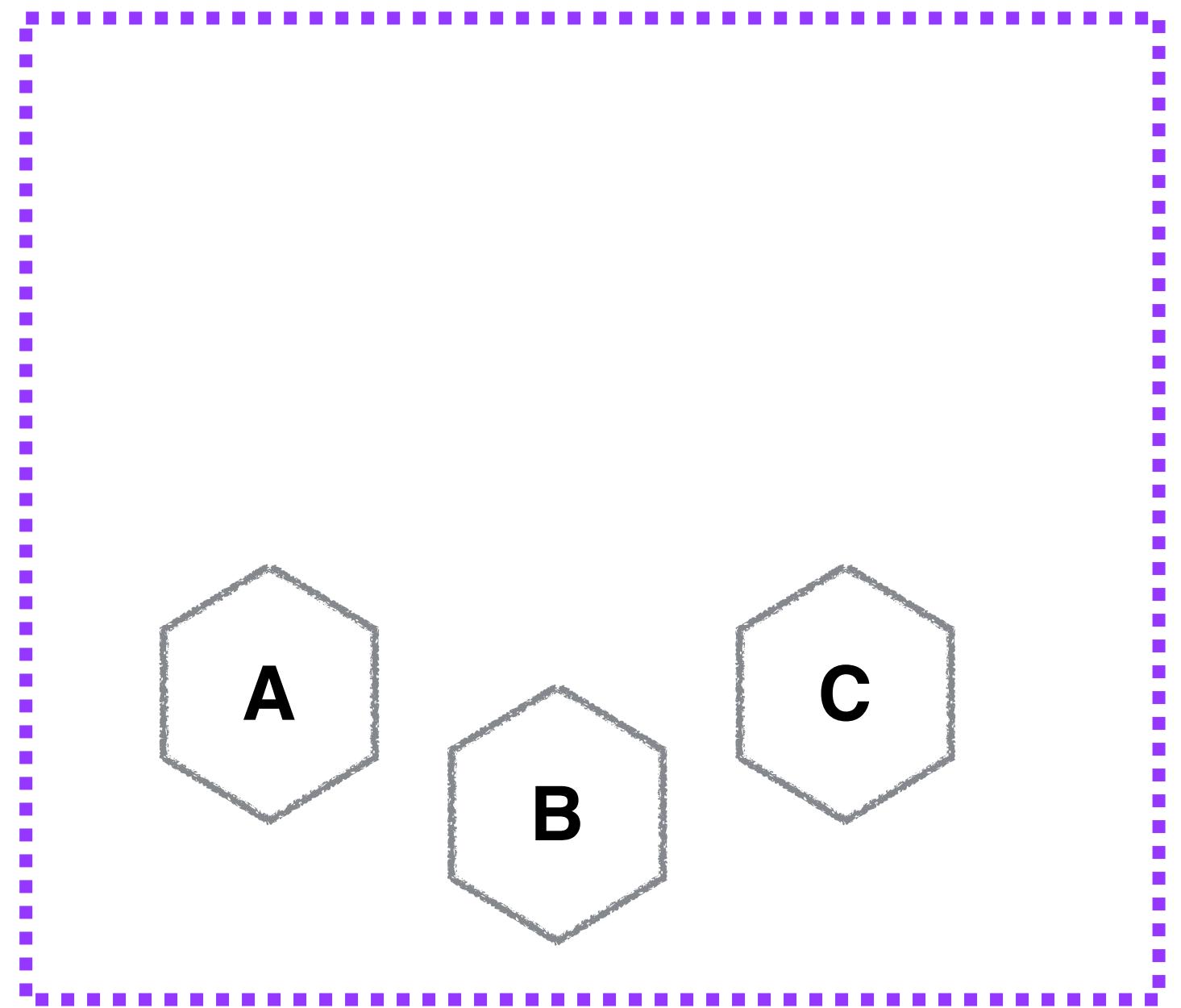
Re-use code across tech stacks

## NETFLIX - SIDECAR PATTERN



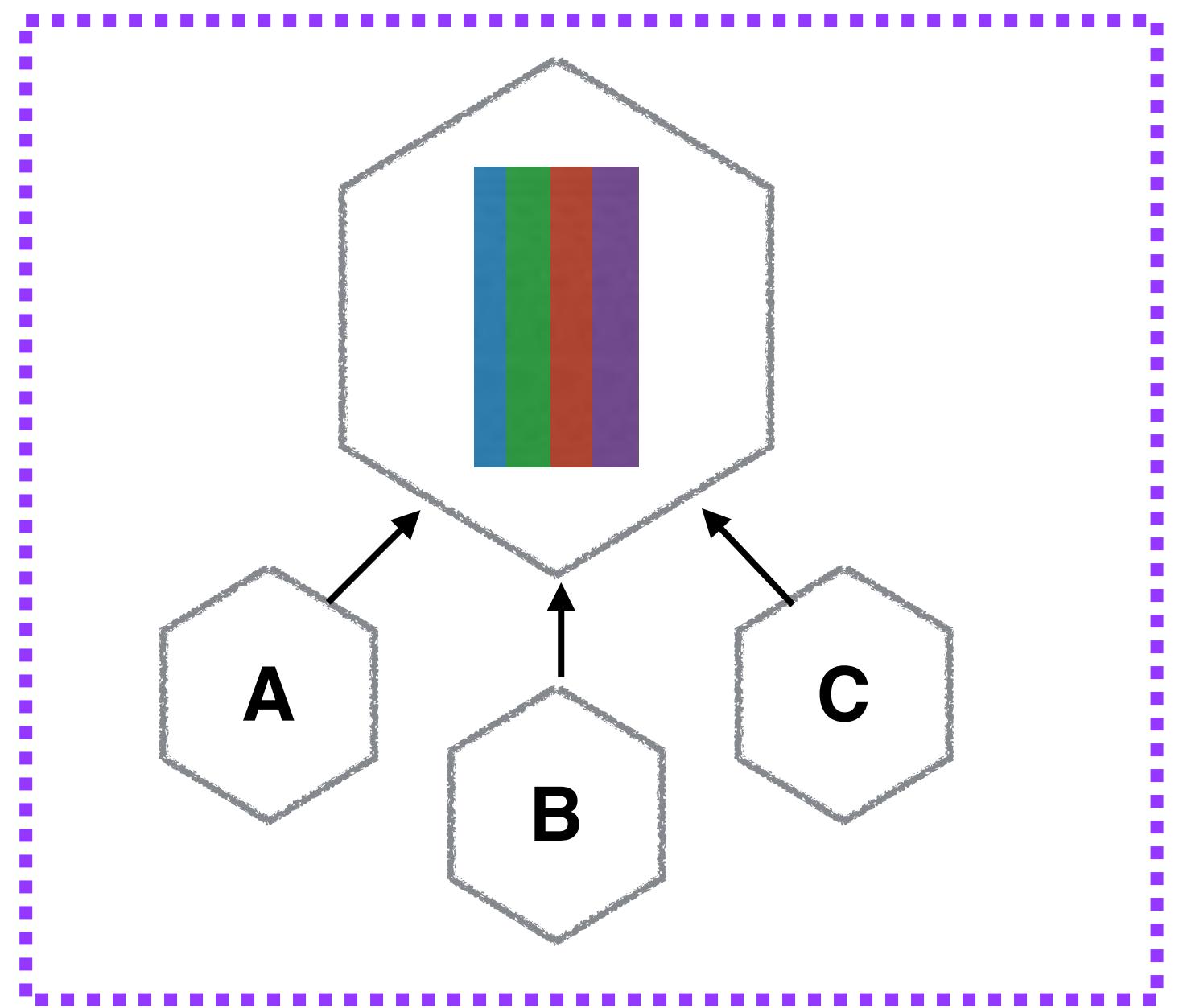
Re-use code across tech stacks  
Reduce impact of version drift

# FROM PROXIES TO SERVICE MESHS



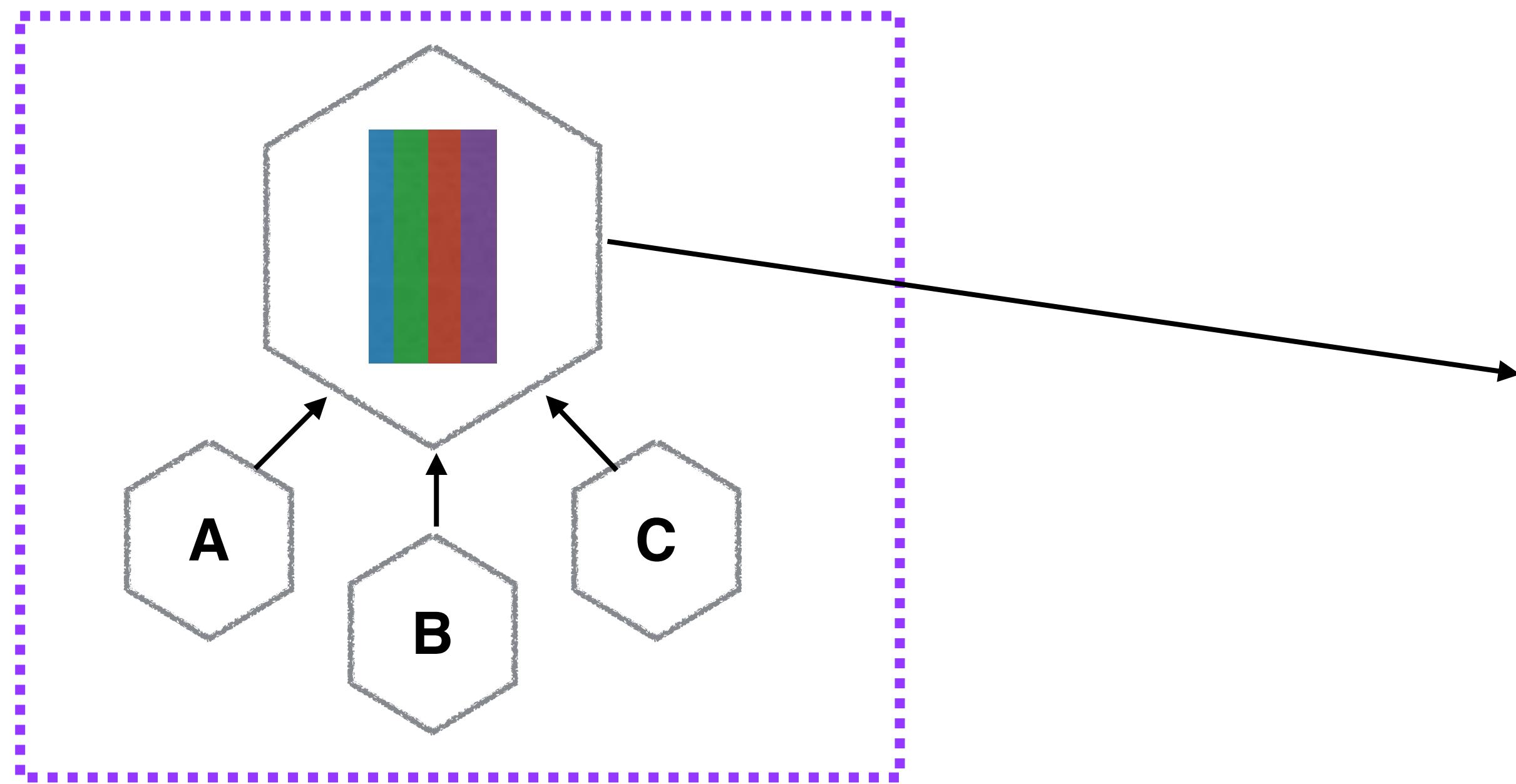
**Machine**

# FROM PROXIES TO SERVICE MESHS



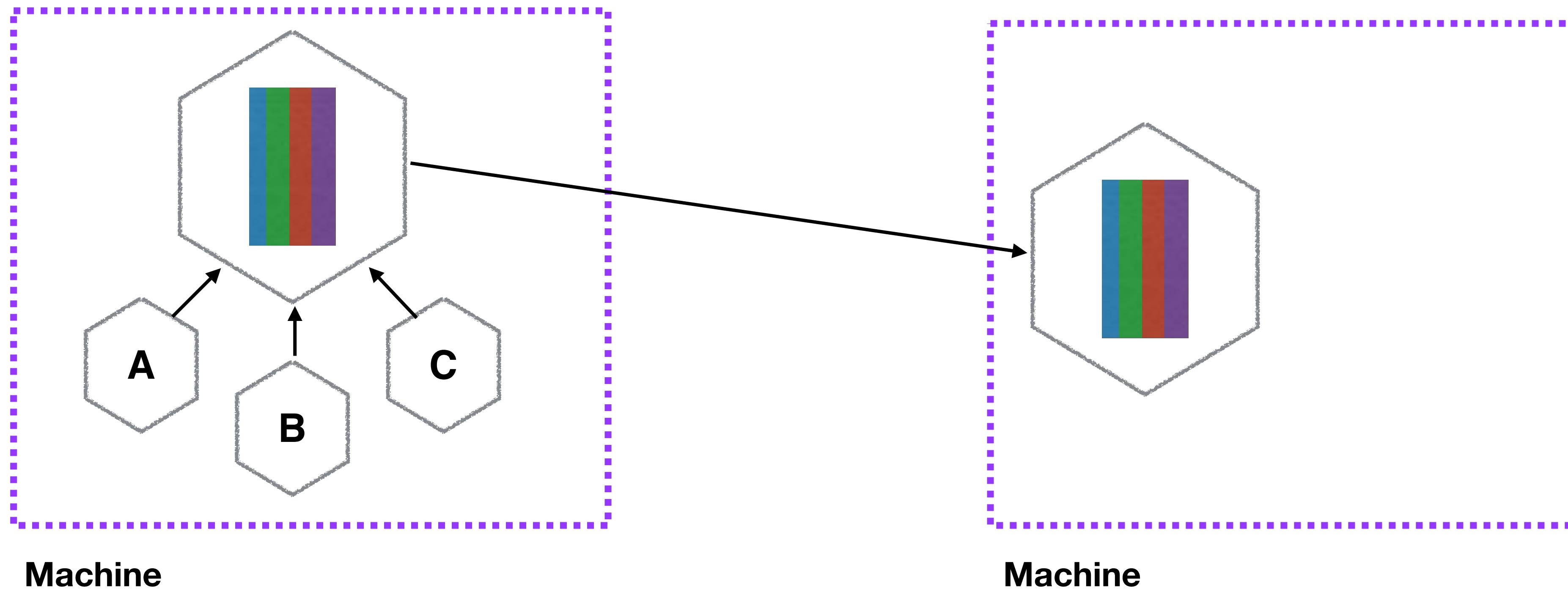
**Machine**

# FROM PROXIES TO SERVICE MESHS

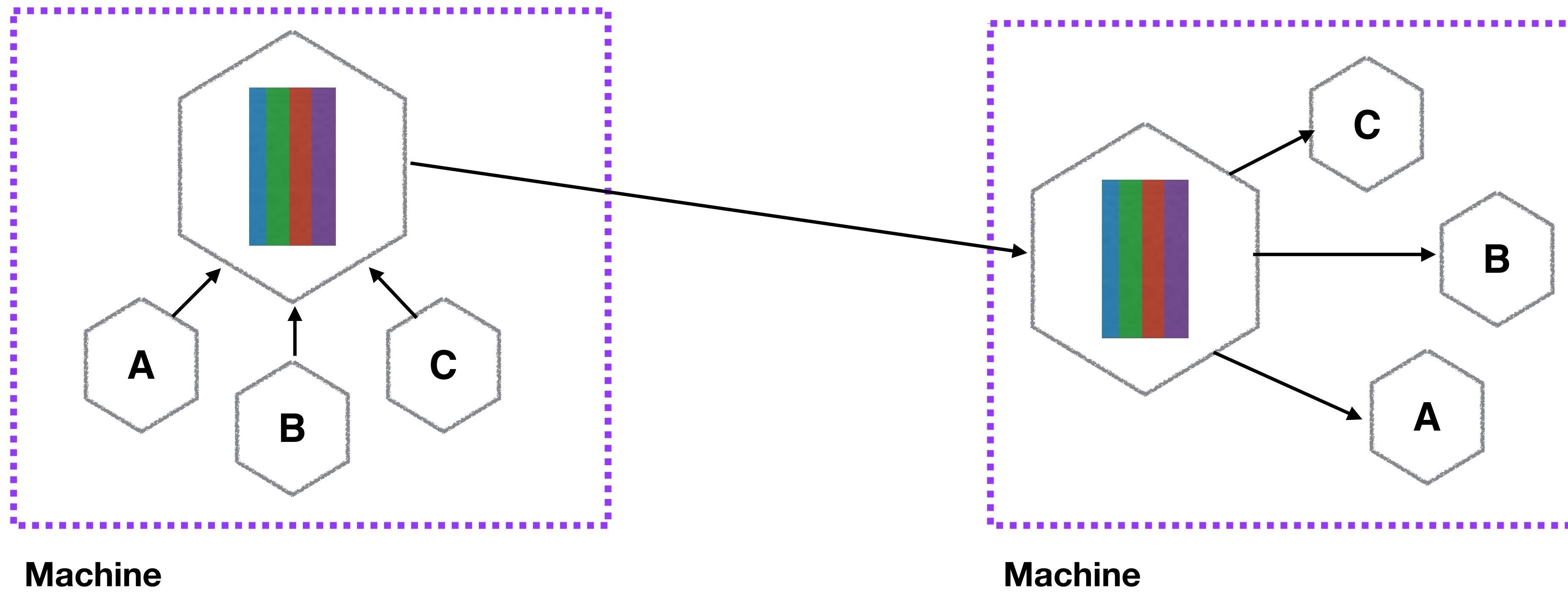


**Machine**

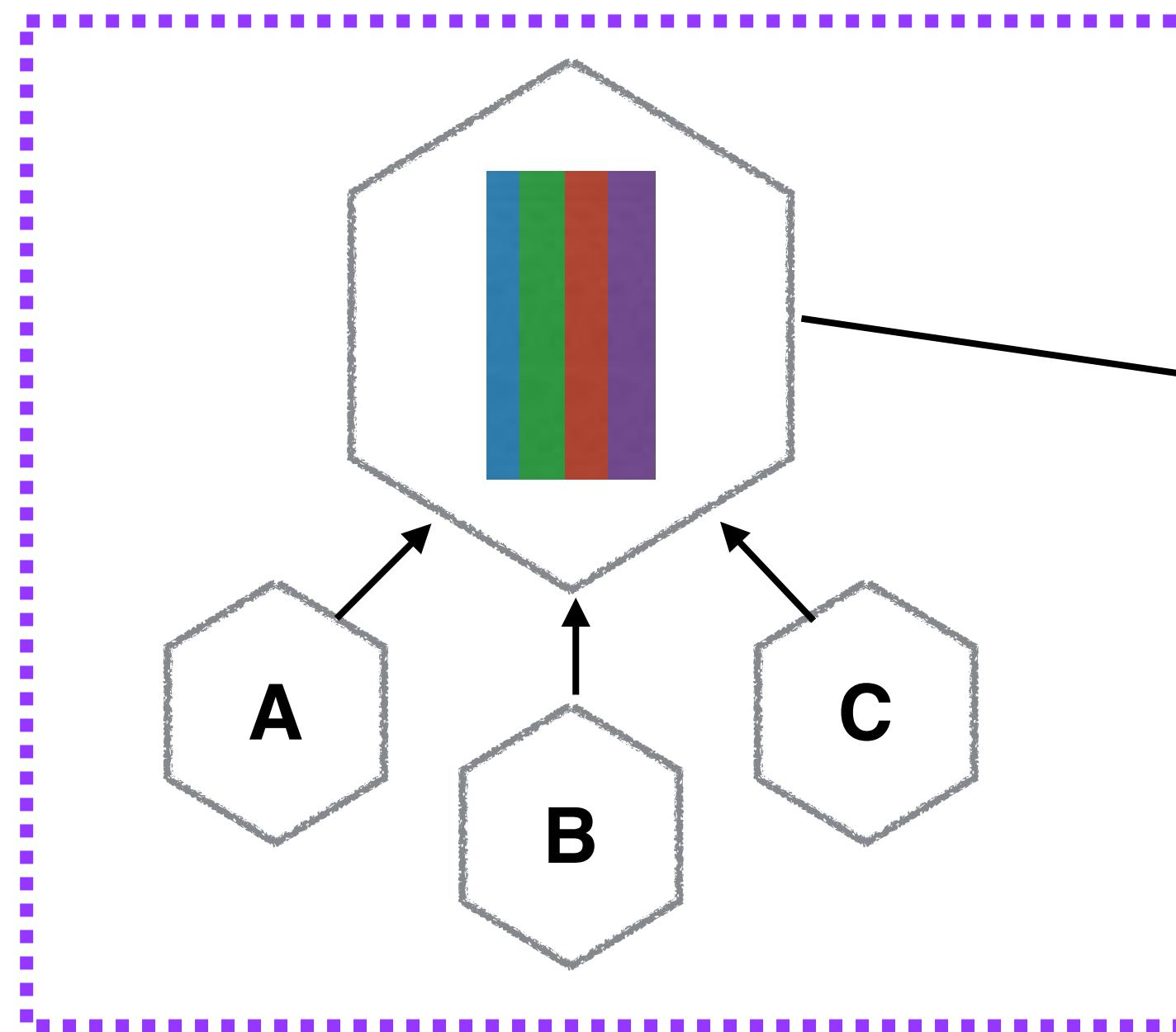
# FROM PROXIES TO SERVICE MESHS



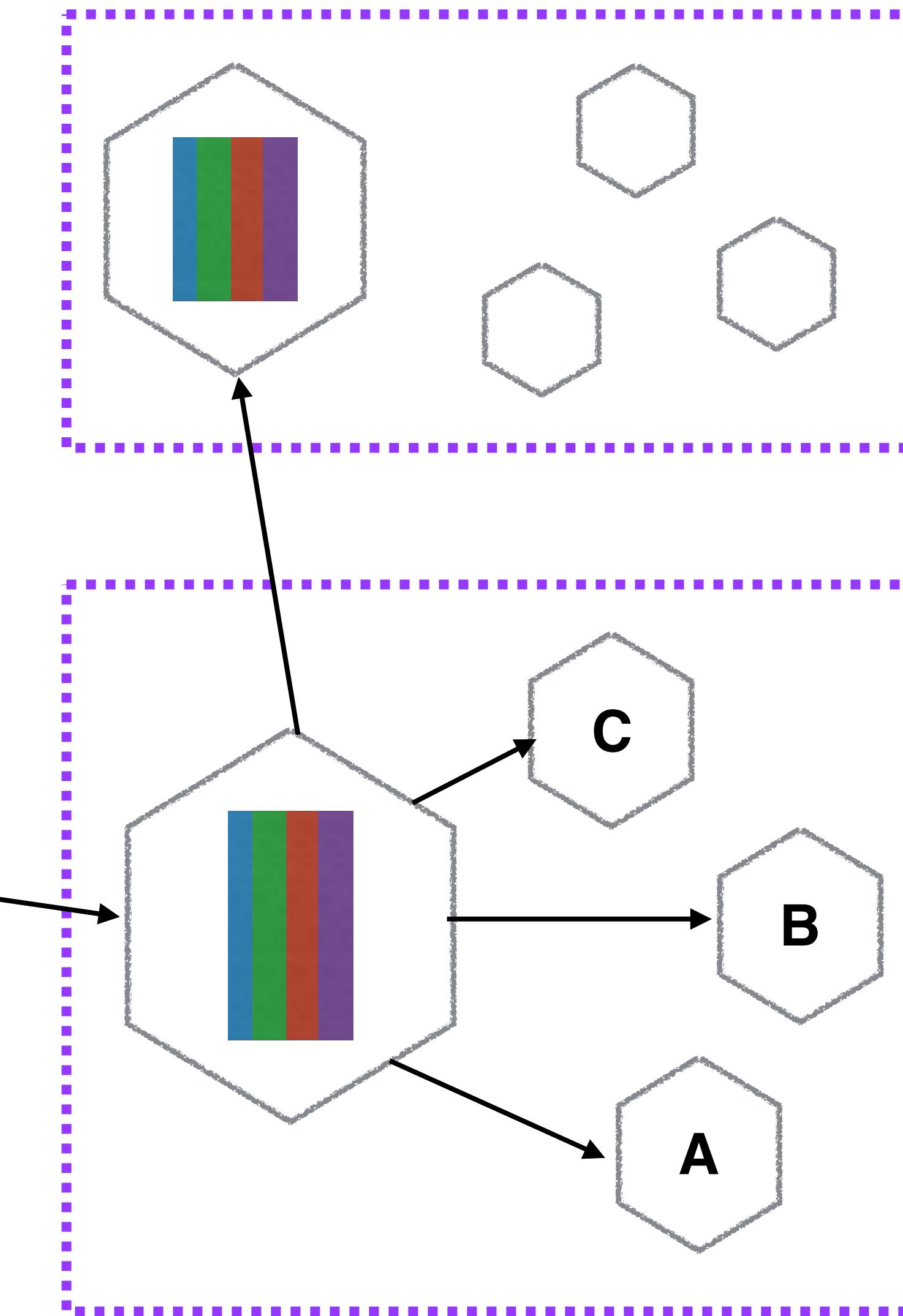
# FROM PROXIES TO SERVICE MESHS



# FROM PROXIES TO SERVICE MESHS

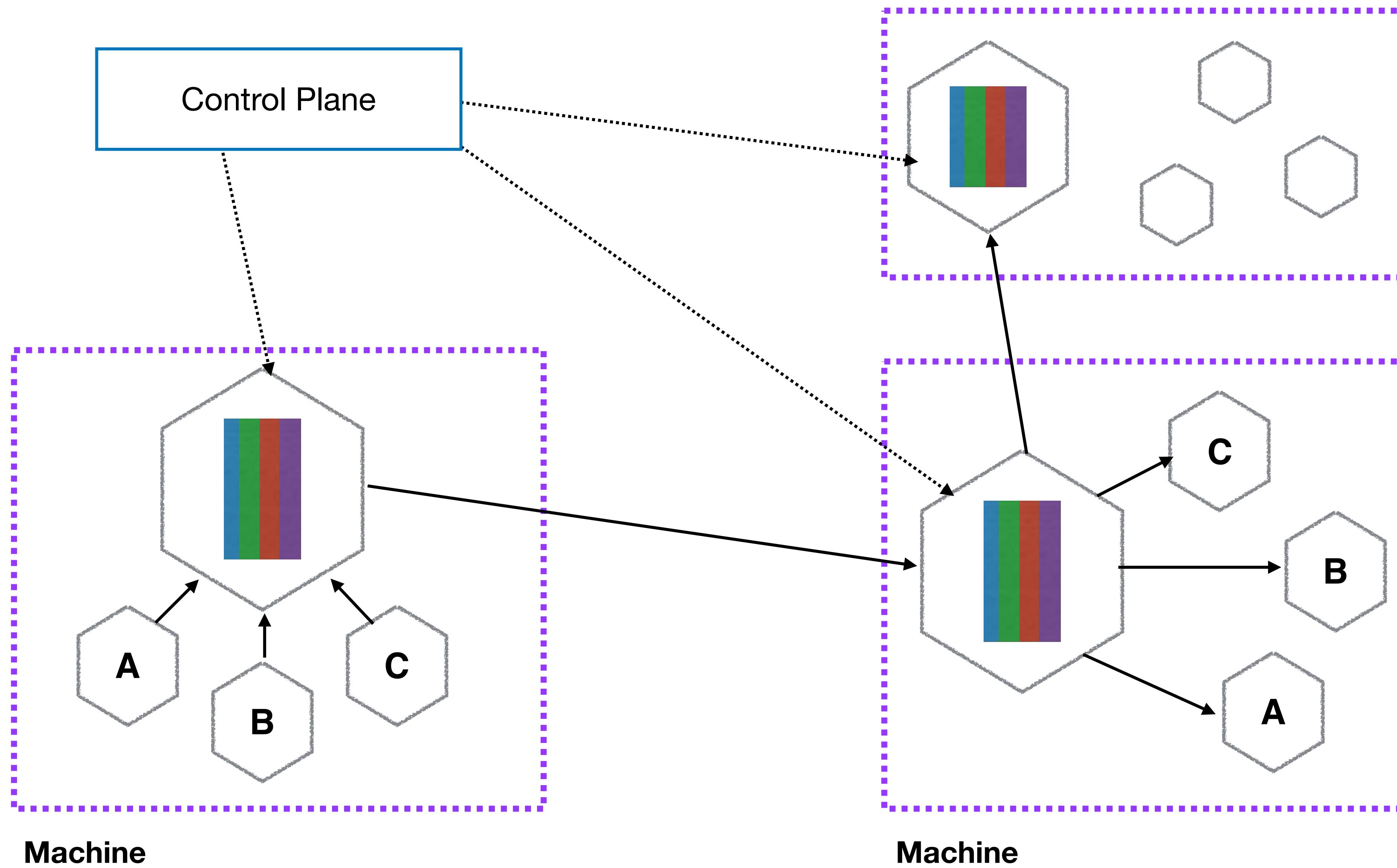


**Machine**



**Machine**

# FROM PROXIES TO SERVICE MESHS



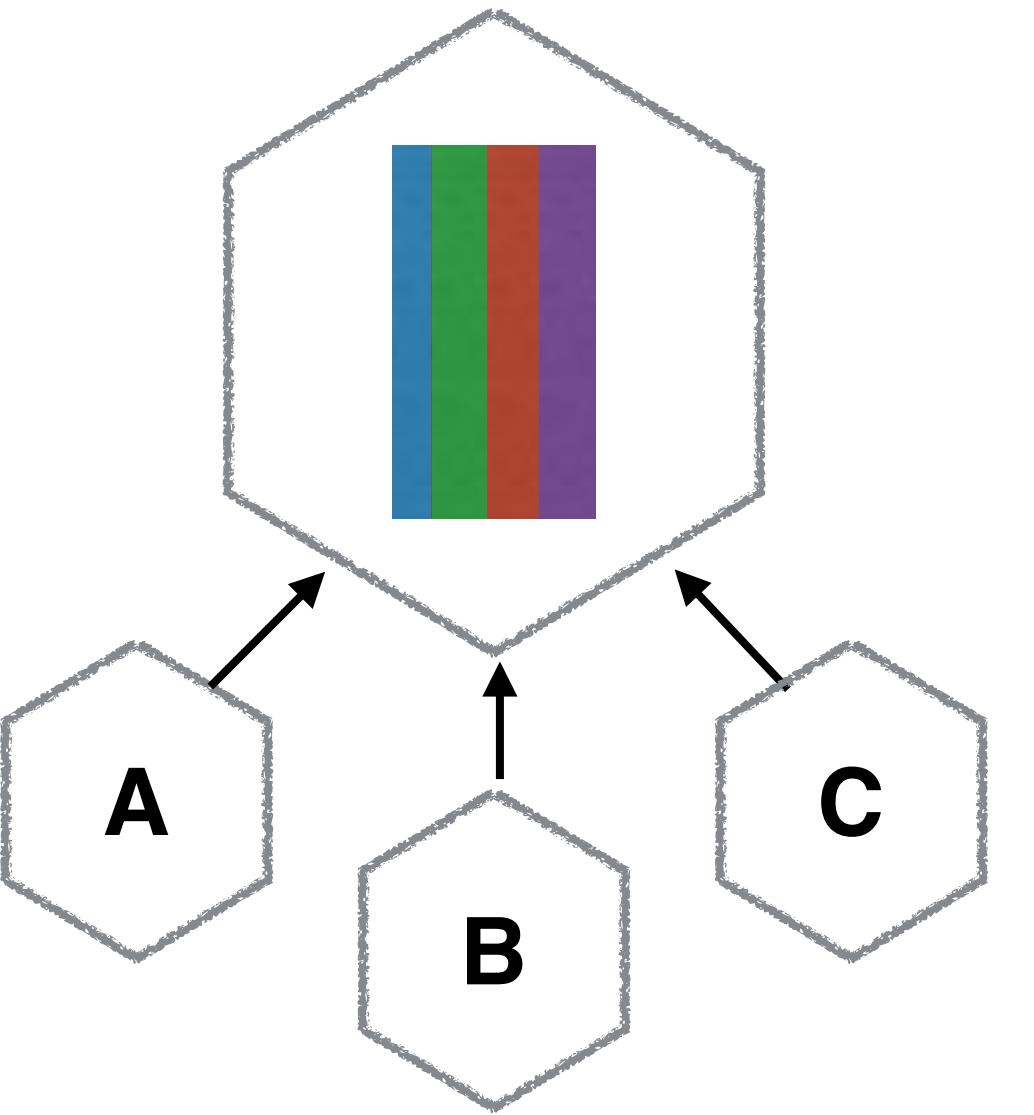
# SIDECARS VS PROXIES

Local Proxy

Sidecar

# SIDECAR VS PROXIES

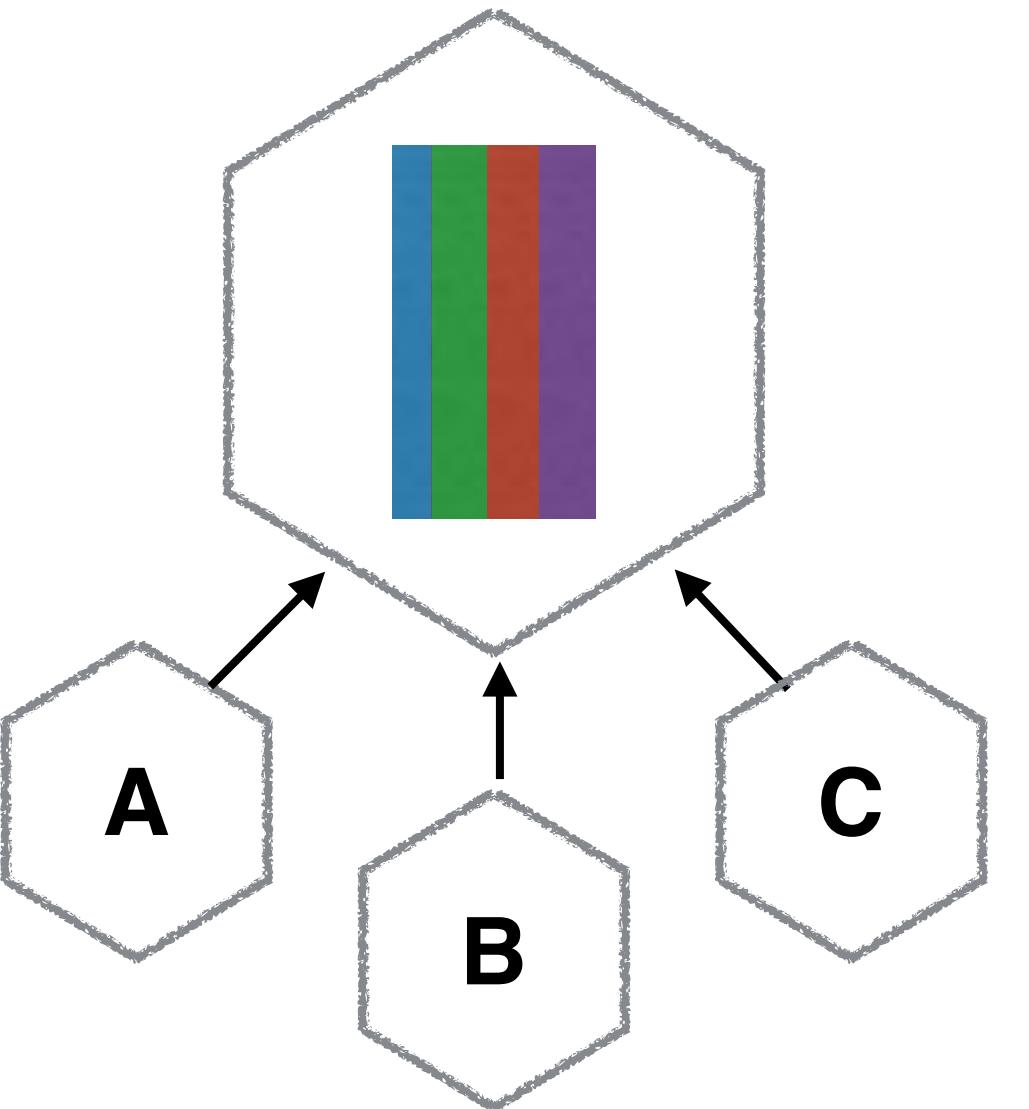
Local Proxy



Sidecar

# SIDECAR VS PROXIES

Local Proxy



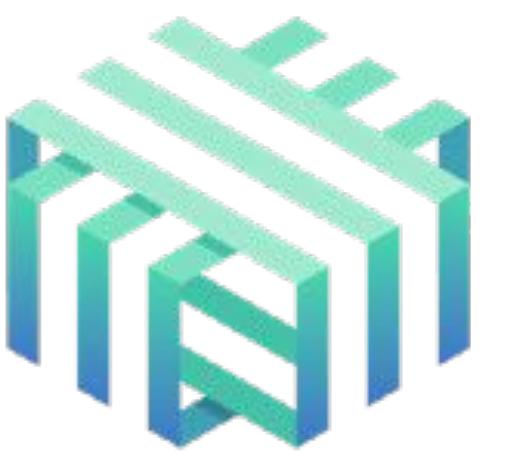
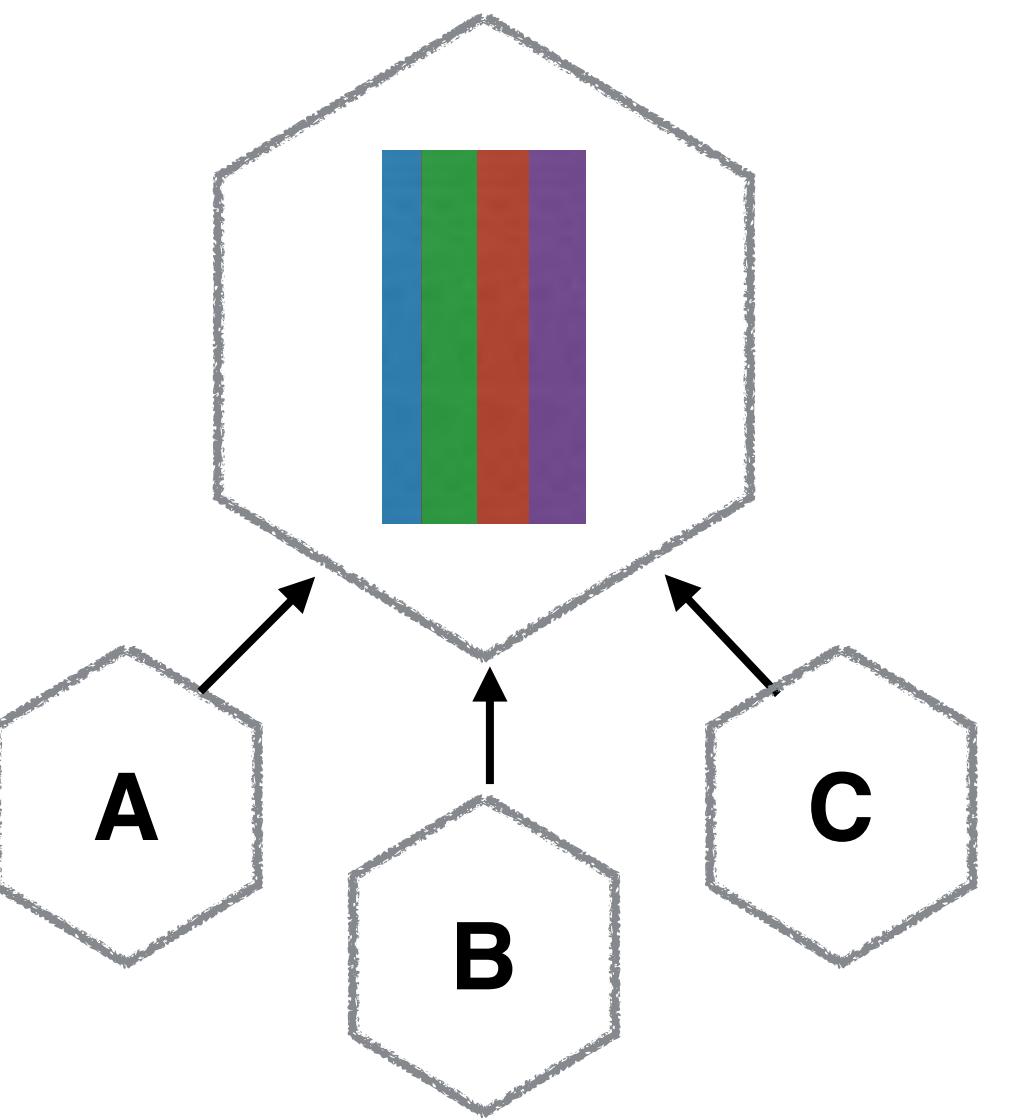
Linkerd

Sidecar



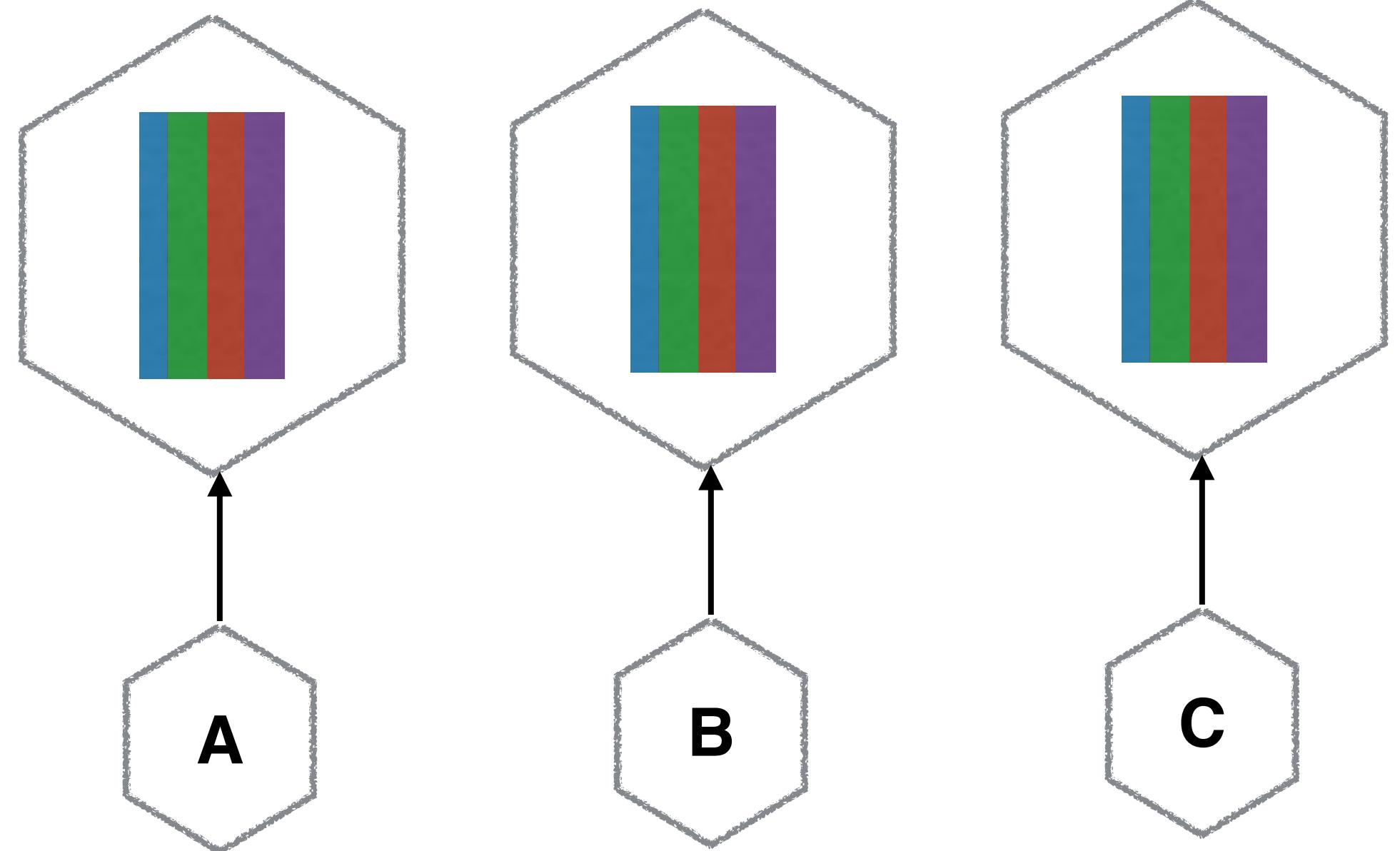
# SIDECAR VS PROXIES

## Local Proxy



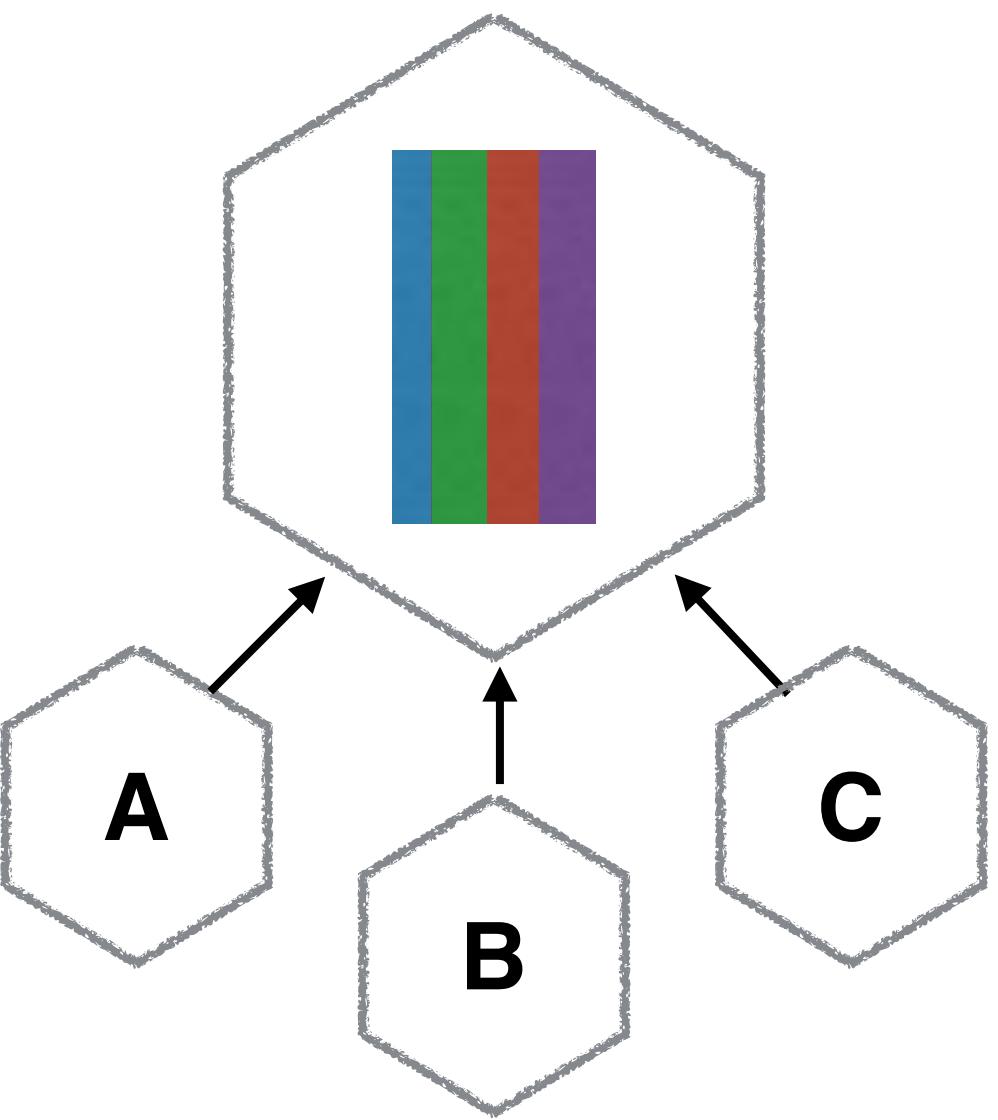
Linkerd

## Sidecar



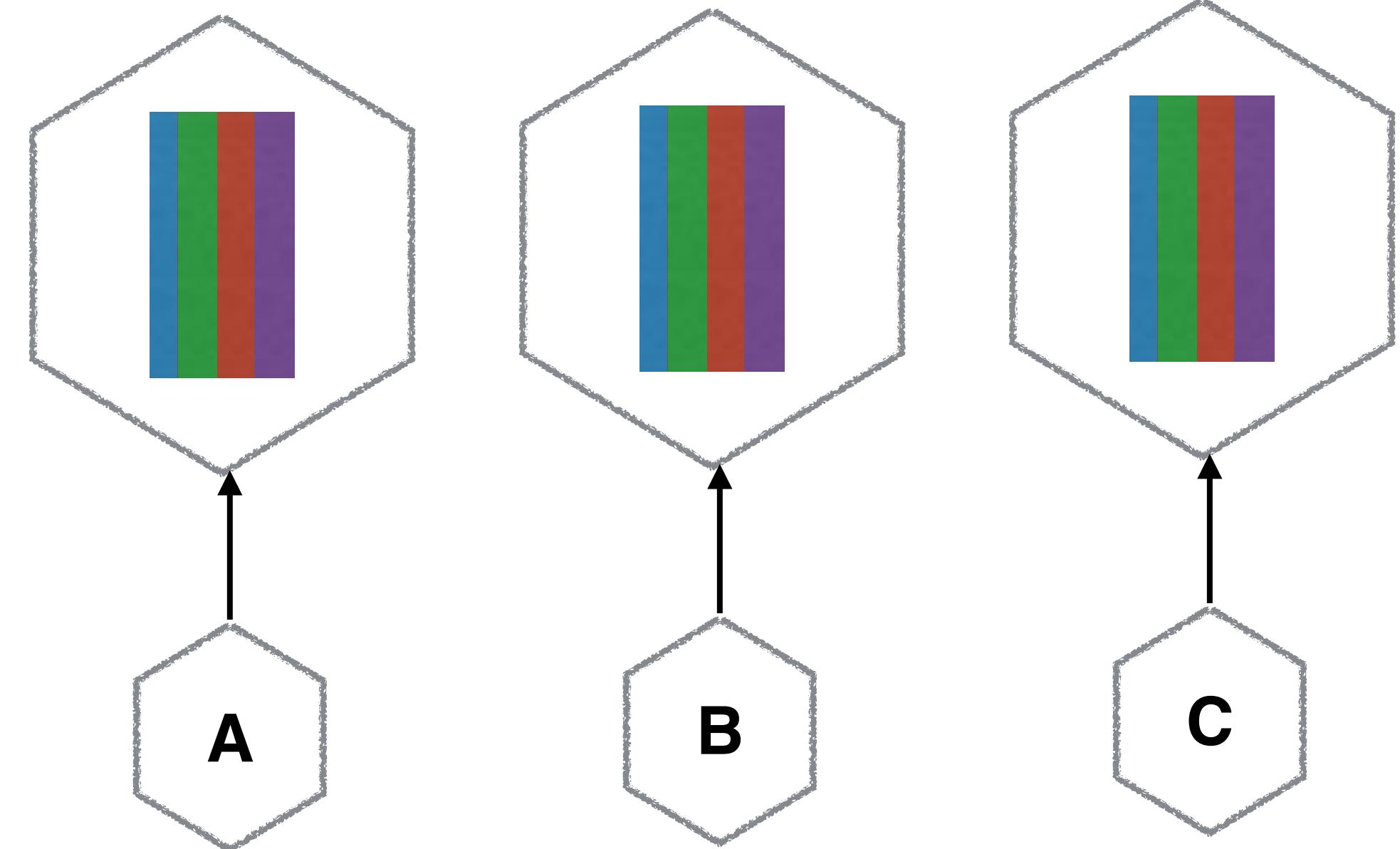
# SIDECAR VS PROXIES

Local Proxy



Linkerd

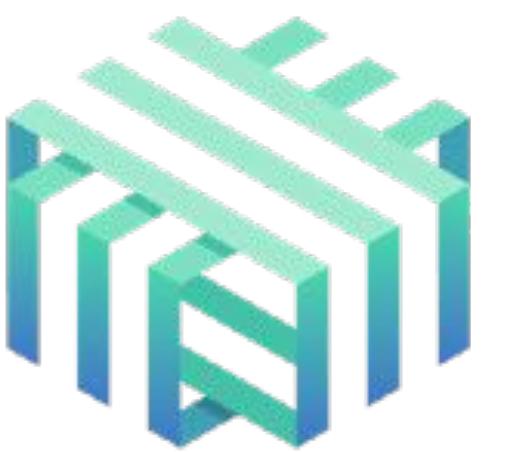
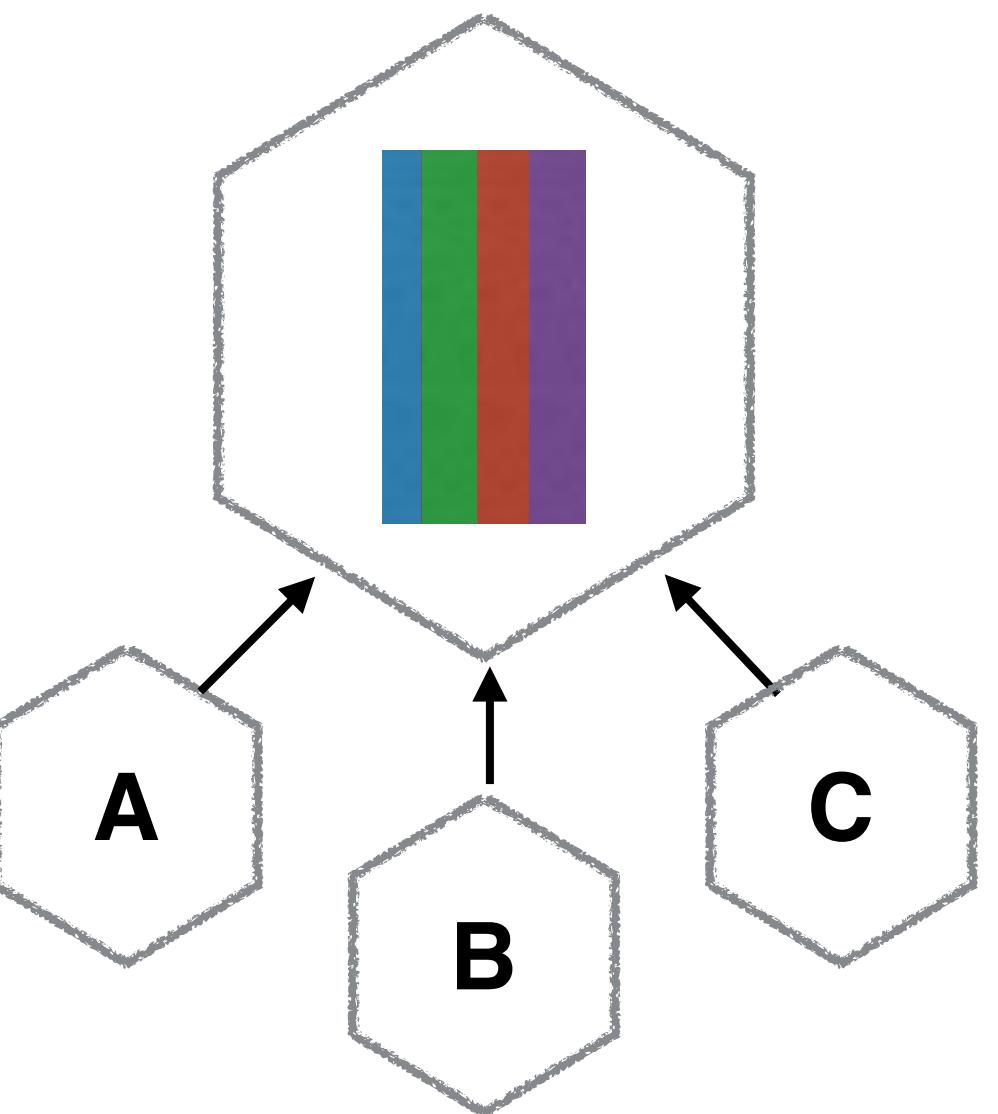
Sidecar



Istio

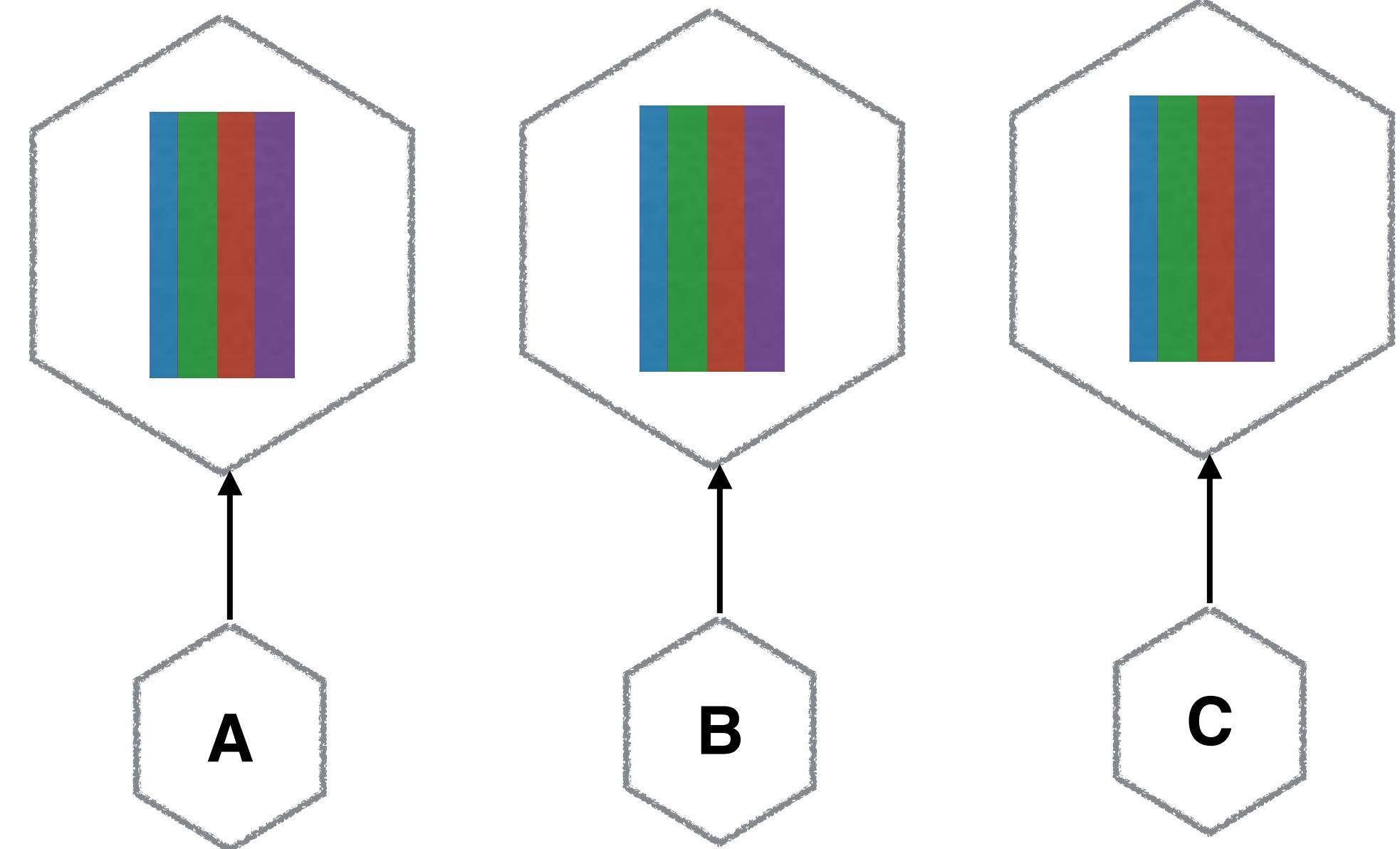
# SIDECAR VS PROXIES

## Local Proxy



Linkerd

## Sidecar



Istio



# SERVICE MESH CAPABILITIES

## SERVICE MESH CAPABILITIES

Load balancing

## SERVICE MESH CAPABILITIES

Load balancing

Traffic Routing (blue/green deploys, canaries)

## SERVICE MESH CAPABILITIES

Load balancing

Traffic Routing (blue/green deploys, canaries)

Service discovery

## SERVICE MESH CAPABILITIES

Load balancing

Traffic Routing (blue/green deploys, canaries)

Service discovery

Tracing

## SERVICE MESH CAPABILITIES

Load balancing

Traffic Routing (blue/green deploys, canaries)

Service discovery

Tracing

Security!

# MUTUAL TLS

## Mutual TLS Authentication

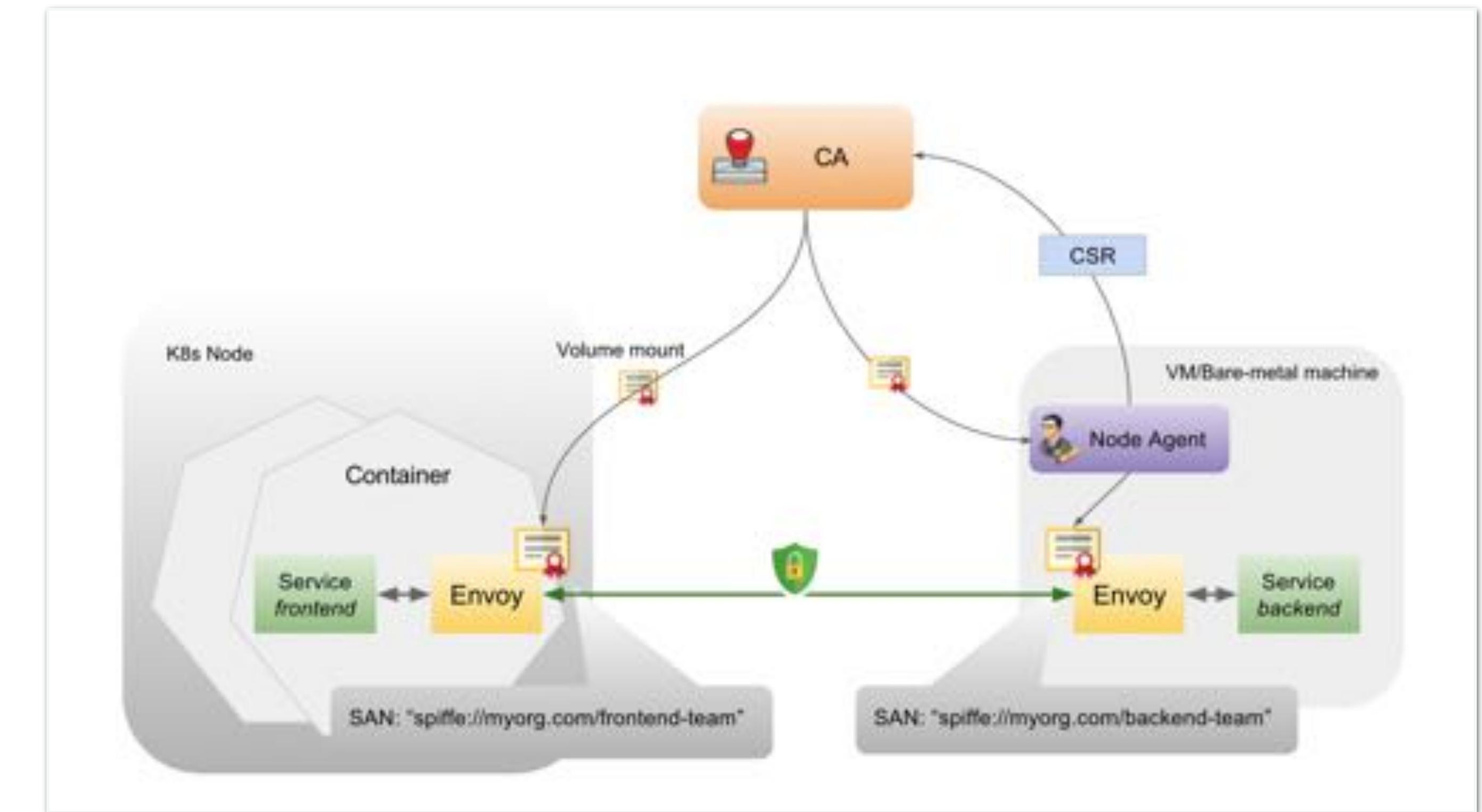
### Overview

Istio Auth's aim is to enhance the security of microservices and their communication without requiring service code changes. It is responsible for:

- Providing each service with a strong identity that represents its role to enable interoperability across clusters and clouds
- Securing service to service communication and end-user to service communication
- Providing a key management system to automate key and certificate generation, distribution, rotation, and revocation

### Architecture

The diagram below shows Istio Auth's architecture, which includes three primary components: identity, key management, and communication security. This diagram describes how Istio Auth is used to secure the service-to-service communication between service 'frontend' running as the service account 'frontend-team' and service 'backend' running as the service account 'backend-team'. Istio supports services running on both Kubernetes containers and VM/bare-metal machines.



<https://istio.io/docs/concepts/security/mutual-tls.html>

# **Caution warranted?**

# SUMMARY

## SUMMARY

# Patching & Passwords

## SUMMARY

Patching & Passwords

Storing Secrets

## SUMMARY

Patching & Passwords

Storing Secrets

Transport Security

## SUMMARY

Patching & Passwords

Storing Secrets

Transport Security

Authorisation

## SUMMARY

Patching & Passwords

Storing Secrets

Transport Security

Authorisation

Service Meshes

**THANKS!**

**Sam Newman.**

Home About **Talks** Events Writing Contact

# Insecure Transit - Microservice Security.

[60min Presentation](#)

**PATCHING MADNESS!**

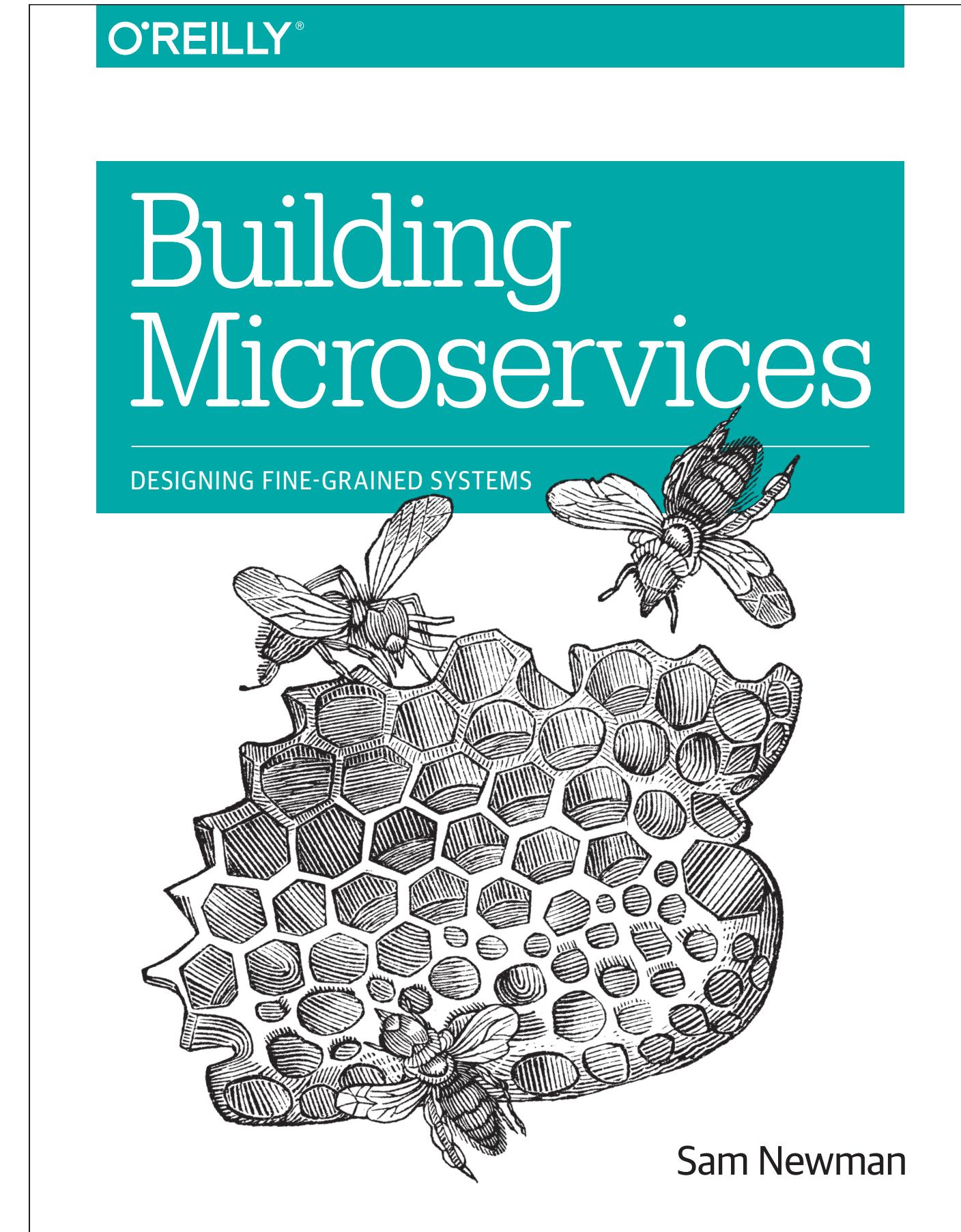
Needs patching

**Book!**

DESIGNING FINE-GRAINED SYSTEMS

I have written a book called "Building Microservices", which is available now. Want to know more? [Read on...](#)

**Video!**



<http://samnewman.io/>

@samnewman