BUS5001 – Cloud Platforms and Analytics

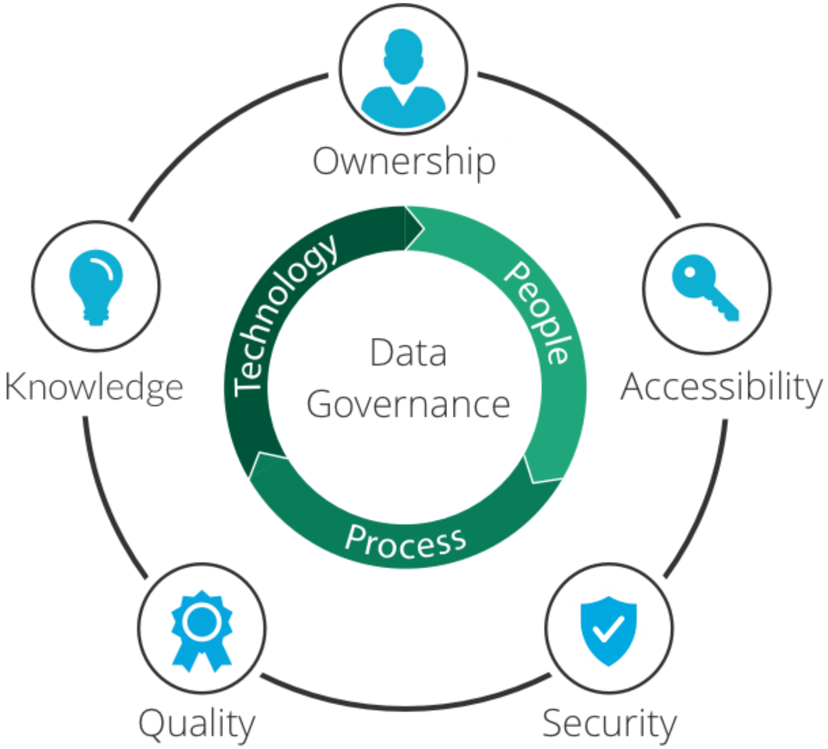
" MAYBE WE SHOULD TRY A DIFFERENT SECURITY APPROACH THIS YEAR. "

Week 05 – Governance, Security, DevOps

Centre for Data Analytics and Cognition
La Trobe University, Australia

Data Governance

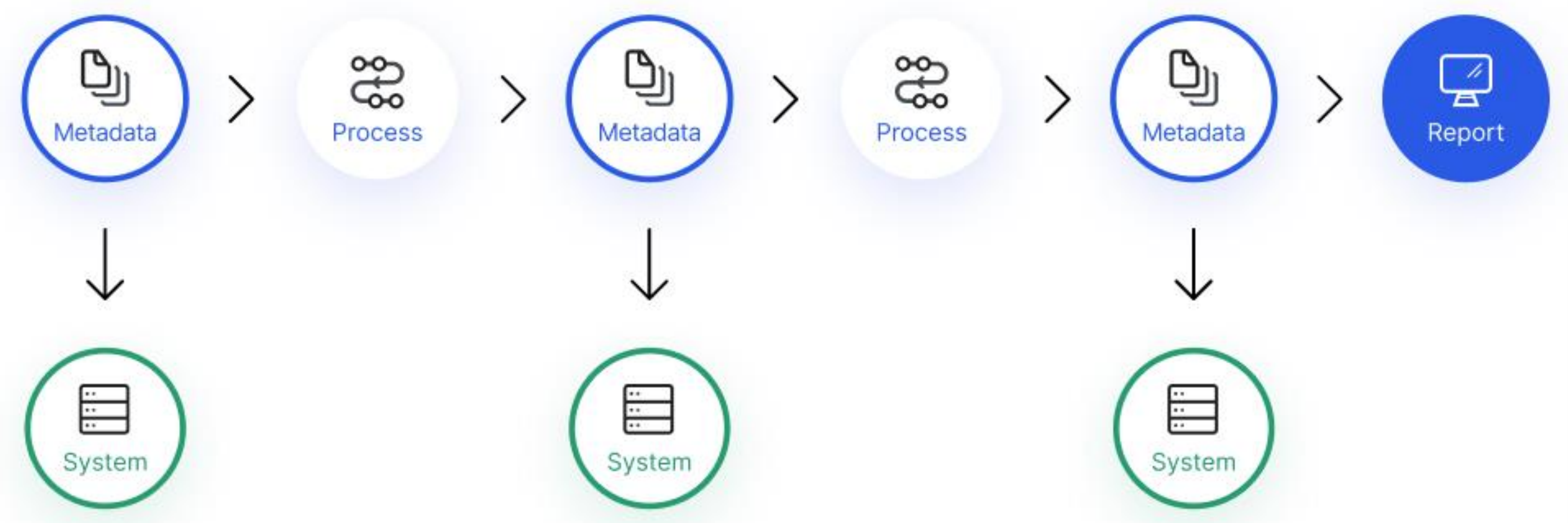
- Data governance is a principled approach to managing data during its life cycle, from acquisition to use to disposal
- Data governance is everything you do to ensure data is secure, private, accurate, available, and usable. It includes the actions people must take, the processes they must follow, and the technology that supports them throughout the data life cycle.



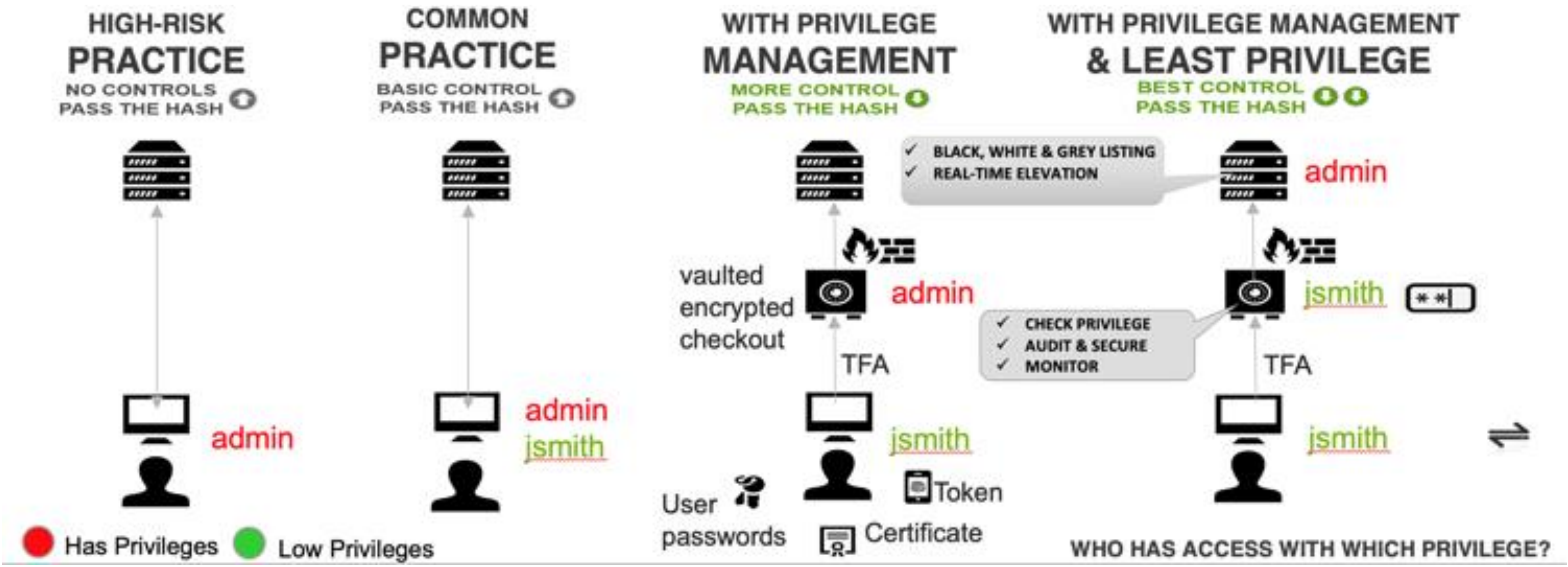
What happens when Governance Fails?

- Yahoo data breach (2013)
    - 3 billion – real names, email addresses, dates of birth, telephone numbers, and security questions
    - $350 million estimated loss in value of company
- First American Financial Corporation data breach (2019)
    - 885 million – bank account numbers, bank statements, mortgage and tax records, social security numbers, wire transaction receipts, and driver license images
    - Poor security
- Equifax data breach (2017)
    - 148 million – Social Security numbers, birth dates, addresses, and in some cases driver license numbers and credit card information
    - $700 million in payouts
- Marriott International data breach (2018)
    - 500 million – combination of contact information, passport number, Starwood Preferred Guest numbers, travel information, credit card numbers and expiration dates, other personal information
    - $24 million, class-action lawsuits filed
- Facebook data breach (2019)
    - 540 million – phone numbers, user names, genders, and locations

Data Lineage



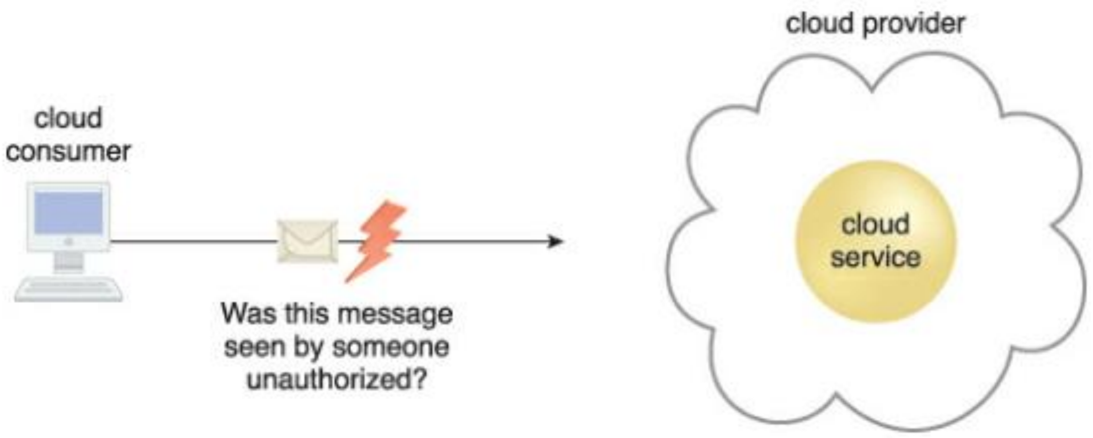
Principle of Least Privilege



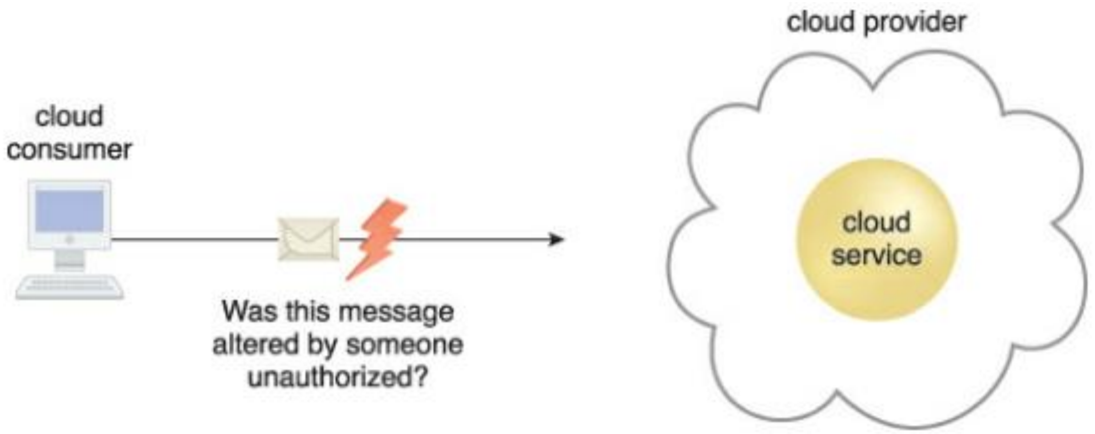
Cloud Security – Terms and Concepts

- Confidentiality
- Integrity
- Authenticity
- Availability
- Threat
- Vulnerability
- Risk
- Security Controls / Mechanisms / Policies

Measuring Security – Confidentiality

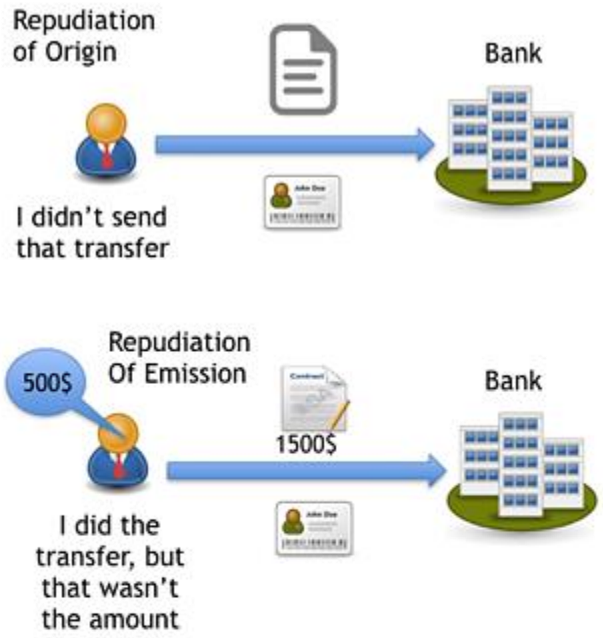


Measuring Security – Integrity



Measuring Security – Authenticity

- Provided by authorised source
- Nonrepudiation – The inability of party to deny or challenge the authenticity of an interaction


Repudiation of Origin
Bank
I didn't send that transfer
500$
Repudiation Of Emission
1500$
Bank
I did the transfer, but that wasn't the amount

Measuring Security – Availability

- Being accessible and usable during a certain time period
- Responsibility generally shared between Cloud Provider and Carrier.
  - If using Cloud to offer services to 3rd Service Consumers then extends to Cloud Consumer

Lack of Security – Threat & Vulnerability

Threat

- A potential security violation that can breach privacy or cause harm
- Generally exploits a known weakness (vulnerability)

Vulnerability

- A weakness that can be exploited because of insufficient security controls or security controls that have been compromised
  - Configurations deficiencies
  - Security policy weakness
  - User errors
  - Hardware / Firmware flaws
  - Software bugs
  - Poor security architecture

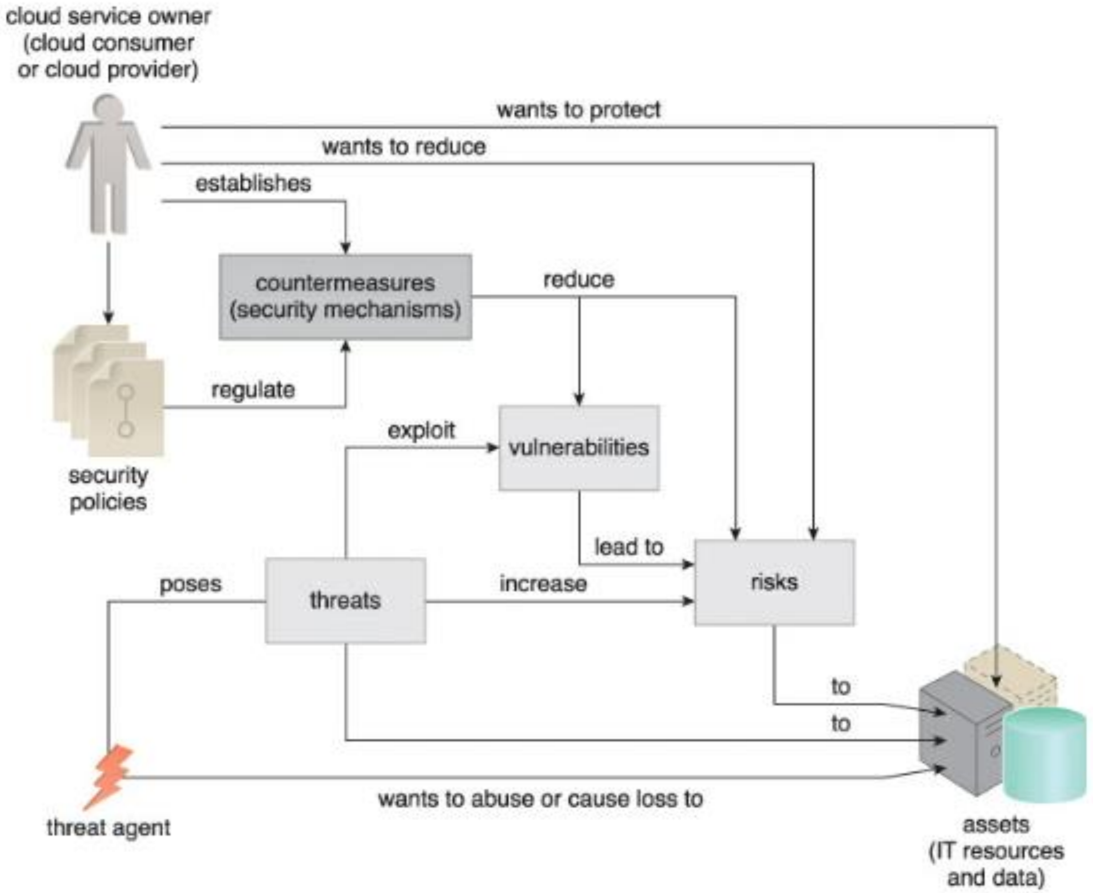
Lack of Security – Risk

- Possibility of loss or harm from an activity
- Measured according to its threat level and the number of possible or known vulnerabilities.
  - The probability of a threat occurring to exploit vulnerabilities in the resource
  - The expectation of loss upon the compromised resource

Improving Security – Controls, Mechanisms, Policies

- Security Controls
  - Countermeasures used to prevent or respond to security threats and to reduce or avoid risk.
- Security Mechanisms
  - Countermeasures are typically described in terms of security mechanisms, which are components comprising a defensive framework
- Security Policies
  - Set of security rules and regulations
  - Definition of how these rules and regulations are implemented and enforced
  - Will direct controls and mechanisms

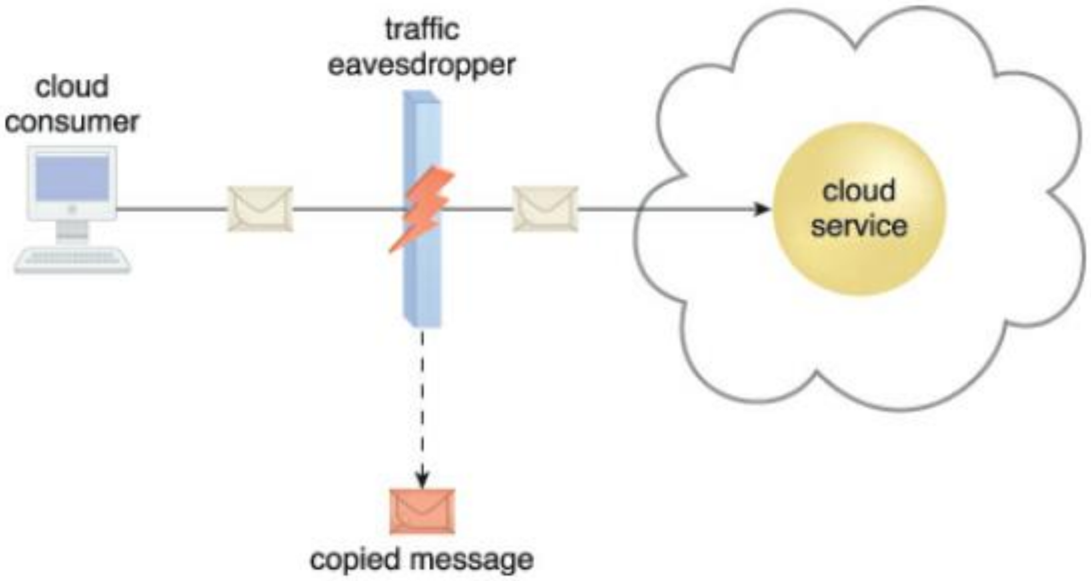
Threat Agents



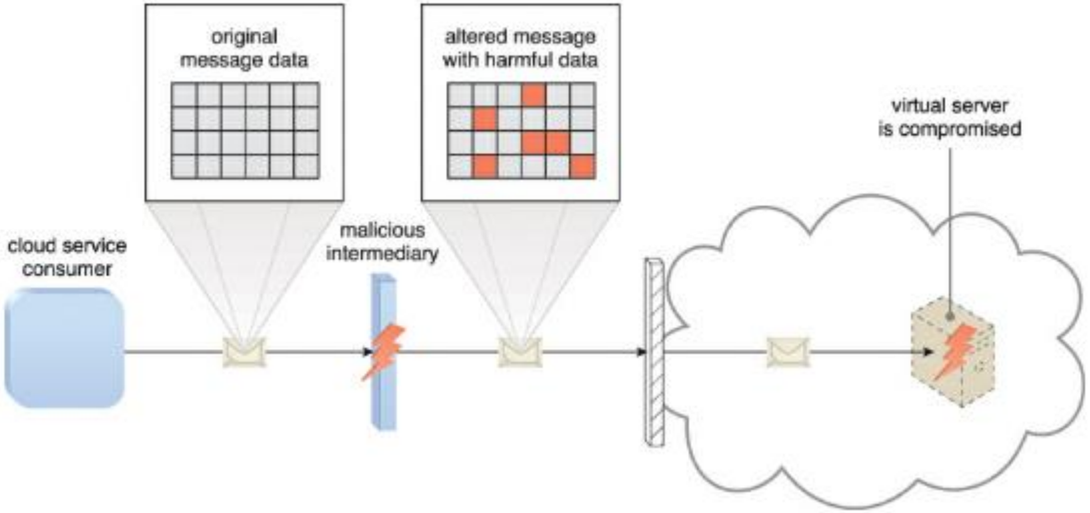
Agents

- Anonymous Attacker
- Malicious Service Agent
- Trusted Attacker
- Malicious Insider

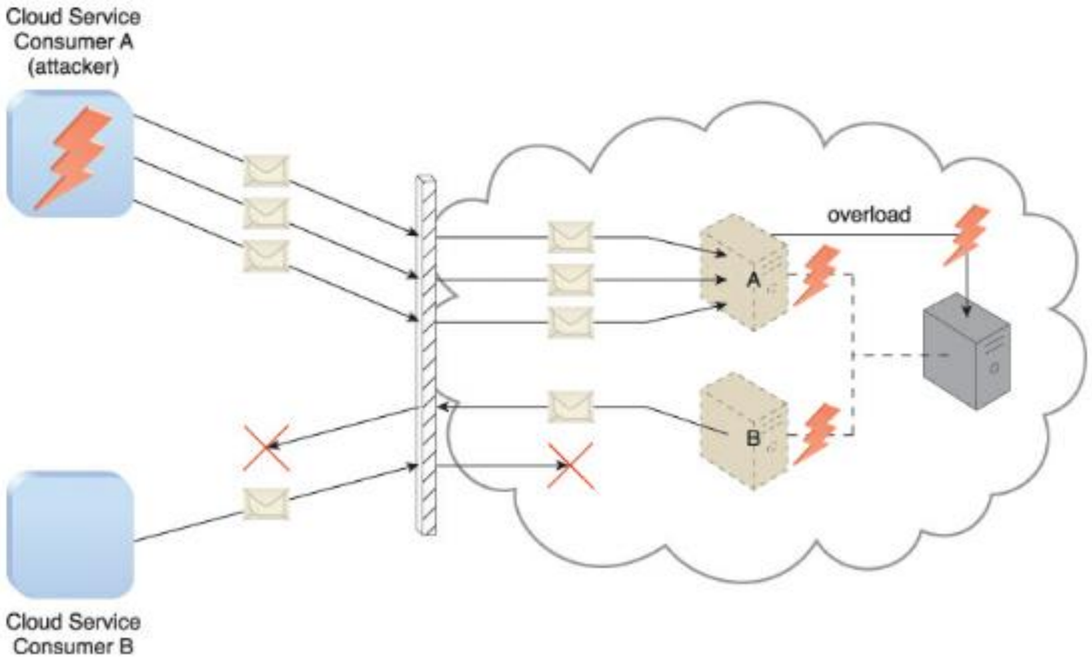
Cloud Security Threats – Traffic Eaves Dropping



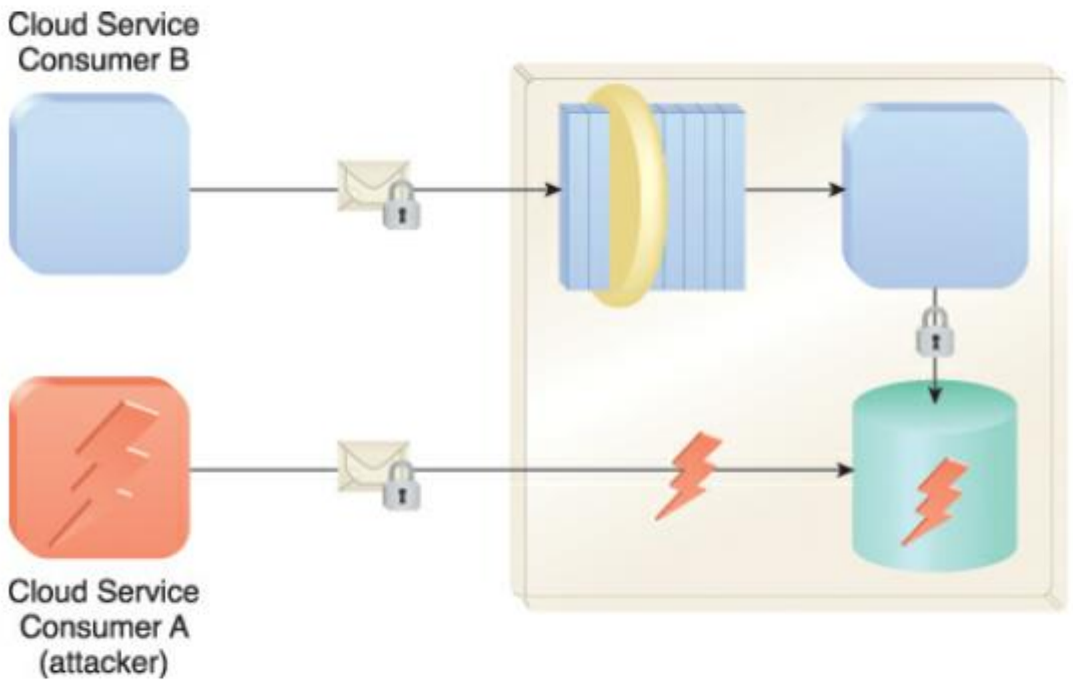
Malicious Intermediary

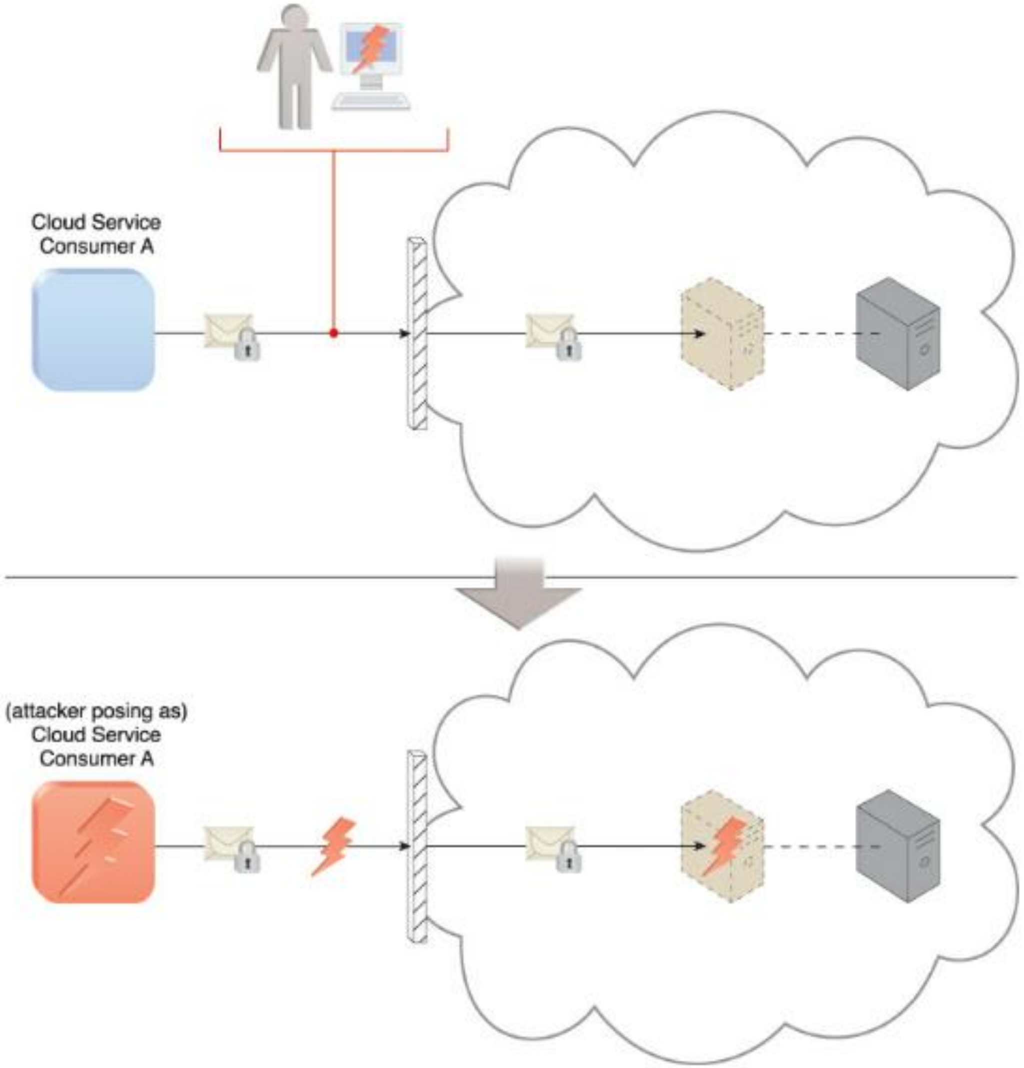


Denial of Service

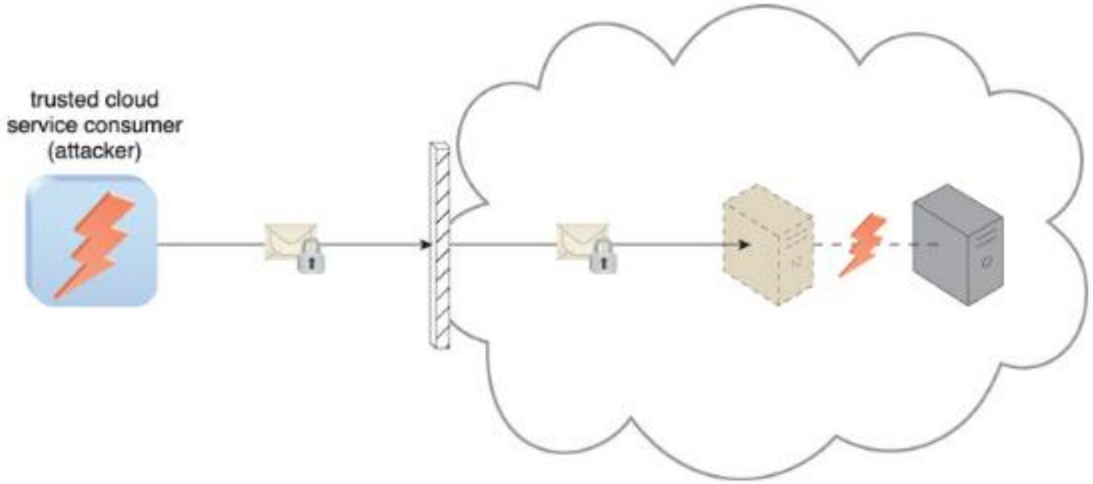


Insufficient Authorisation

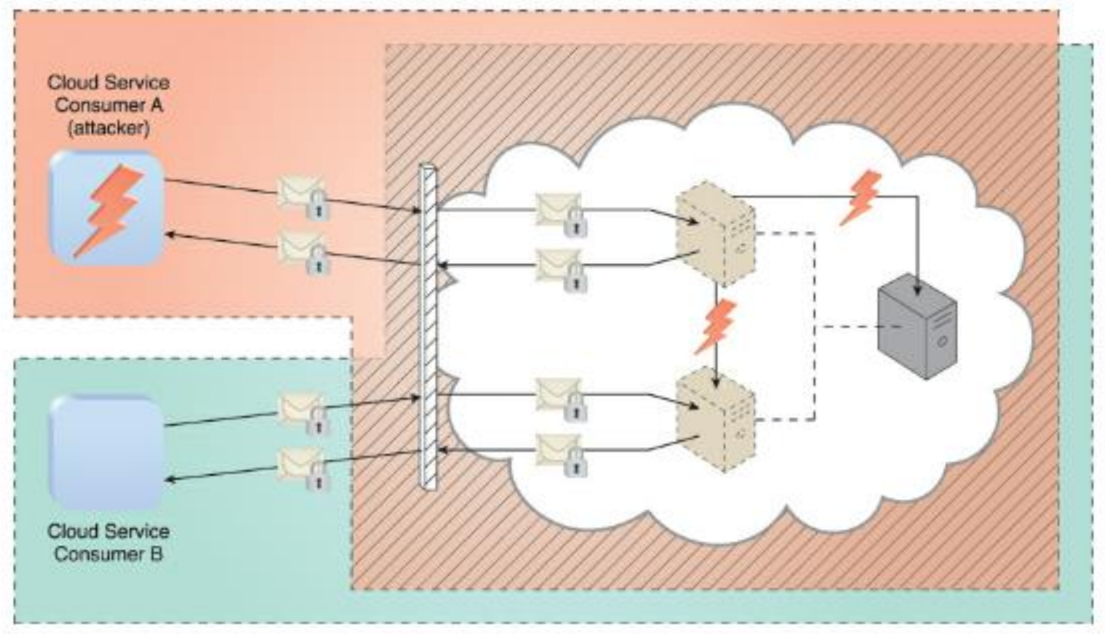



Cloud Service
Consumer A
(attacker posing as)
Cloud Service
Consumer A

Virtualisation Attack



Overlapping Trust Boundaries

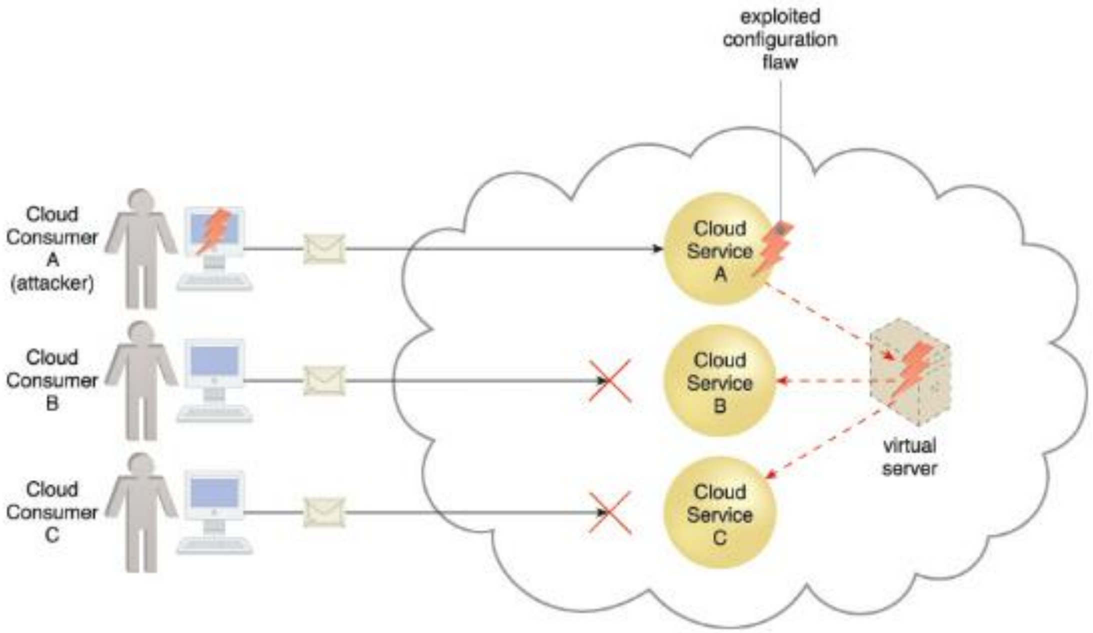


Container Attacks

- Containerisation introduces a lack of isolation from the host operating system
  - Create containers within the OS so that if compromised only those containers have issues
  - One service per physical server model – Reduces risk but increases resource usage thus cost and complexity

Other considerations

- Poorly implemented cloud service deployment

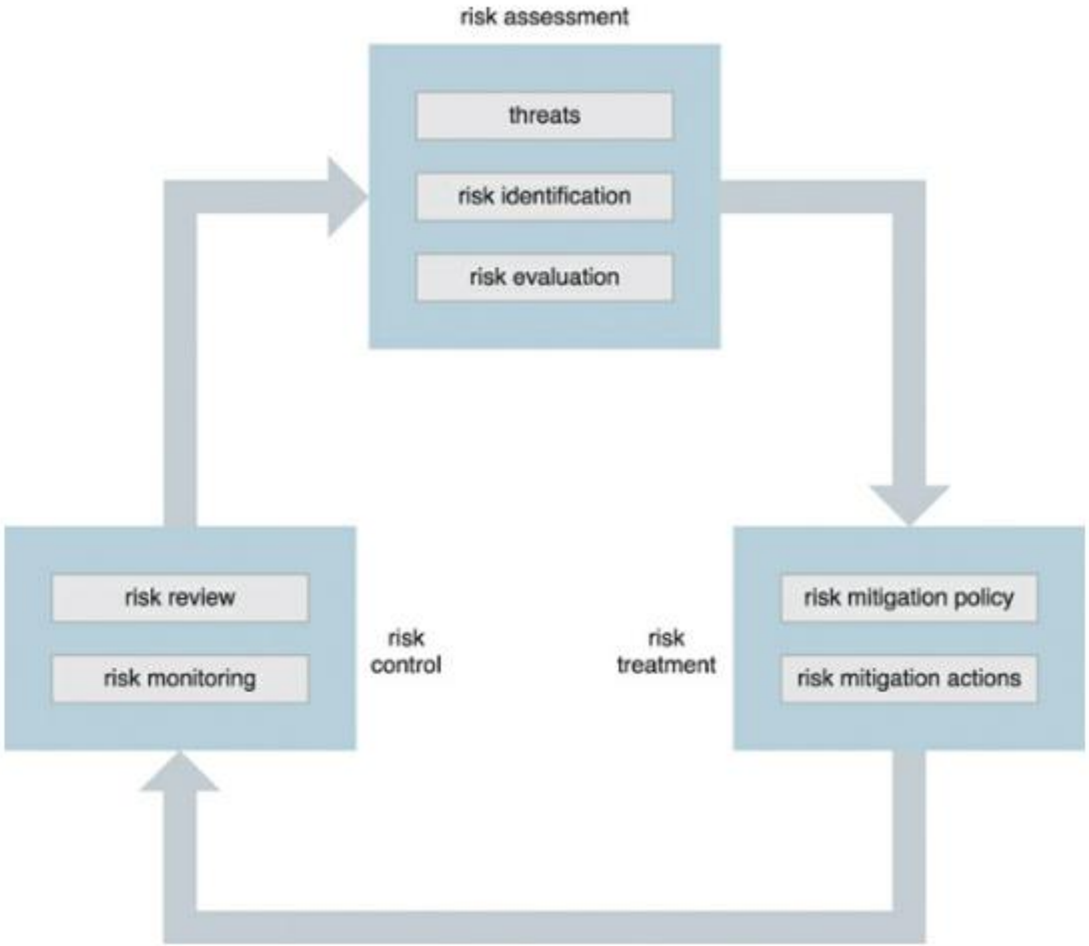


Other considerations

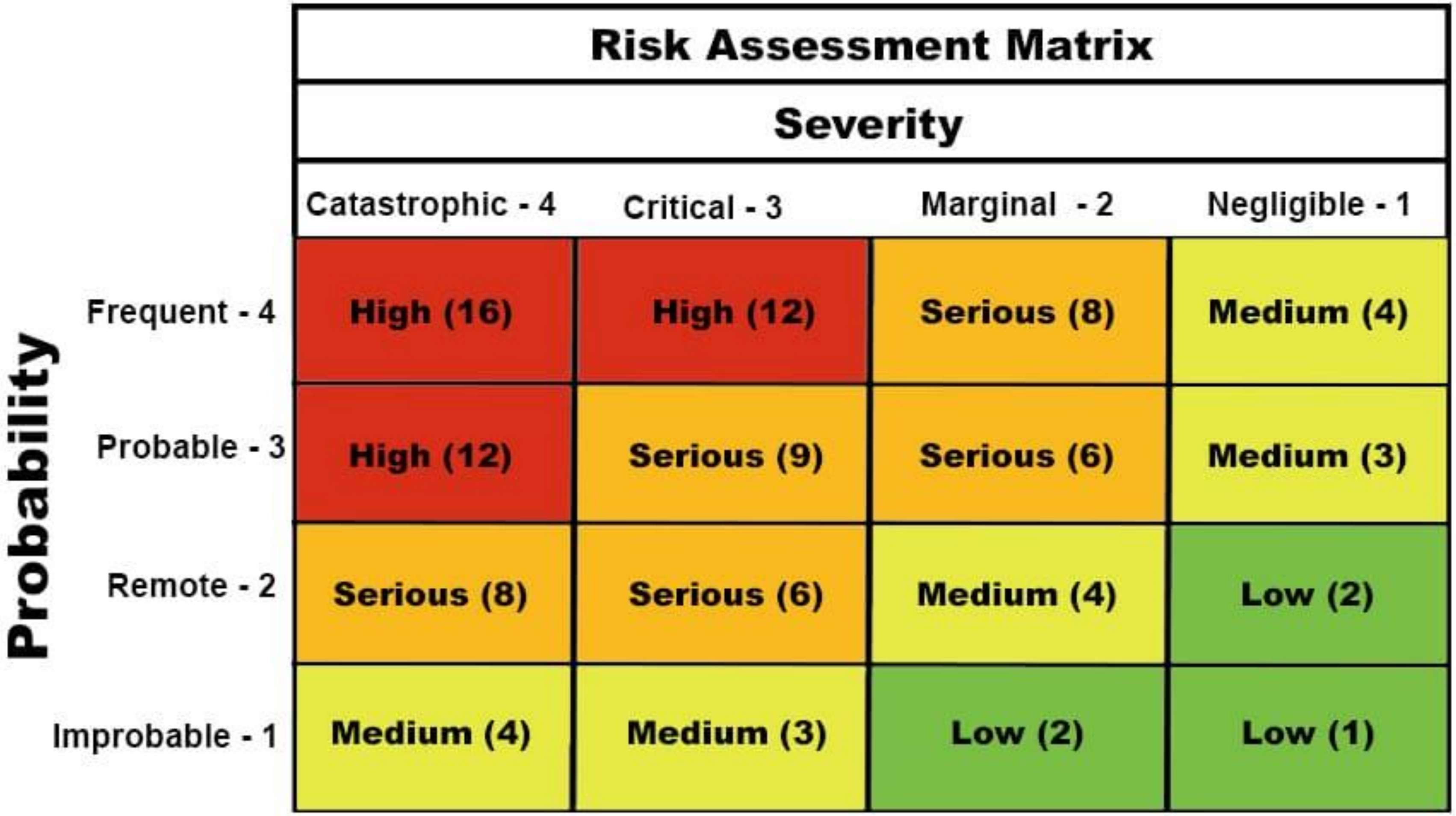
- Incompatibilities in security policies
- Cloud contracts
  - Where does the liability lie?
  - What is the level of indemnity?
  - The greater the risk absorbed by the Provider the less to bear by the consumer
  - What are the lines responsibilities in a shared model? IaaS

Risk Management

- Probability of Occurrence
- Impact of Occurrence



Risk Assessment Matrix



Risk Assessment Matrix

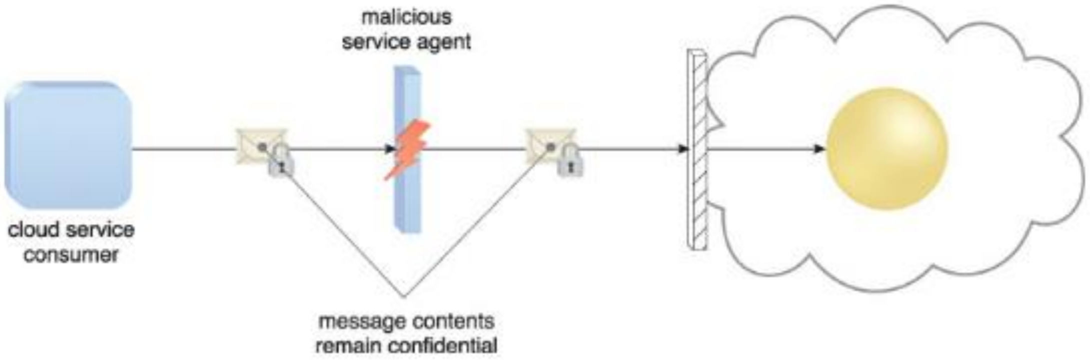
Severity

		Catastrophic - 4	Critical - 3	Marginal - 2	Negligible - 1
Probability	Frequent - 4	**High (16)**	**High (12)**	**Serious (8)**	**Medium (4)**
	Probable - 3	**High (12)**	**Serious (9)**	**Serious (6)**	**Medium (3)**
	Remote - 2	**Serious (8)**	**Serious (6)**	**Medium (4)**	**Low (2)**
	Improbable - 1	**Medium (4)**	**Medium (3)**	**Low (2)**	**Low (1)**

Cloud Security – Security Mechanisms

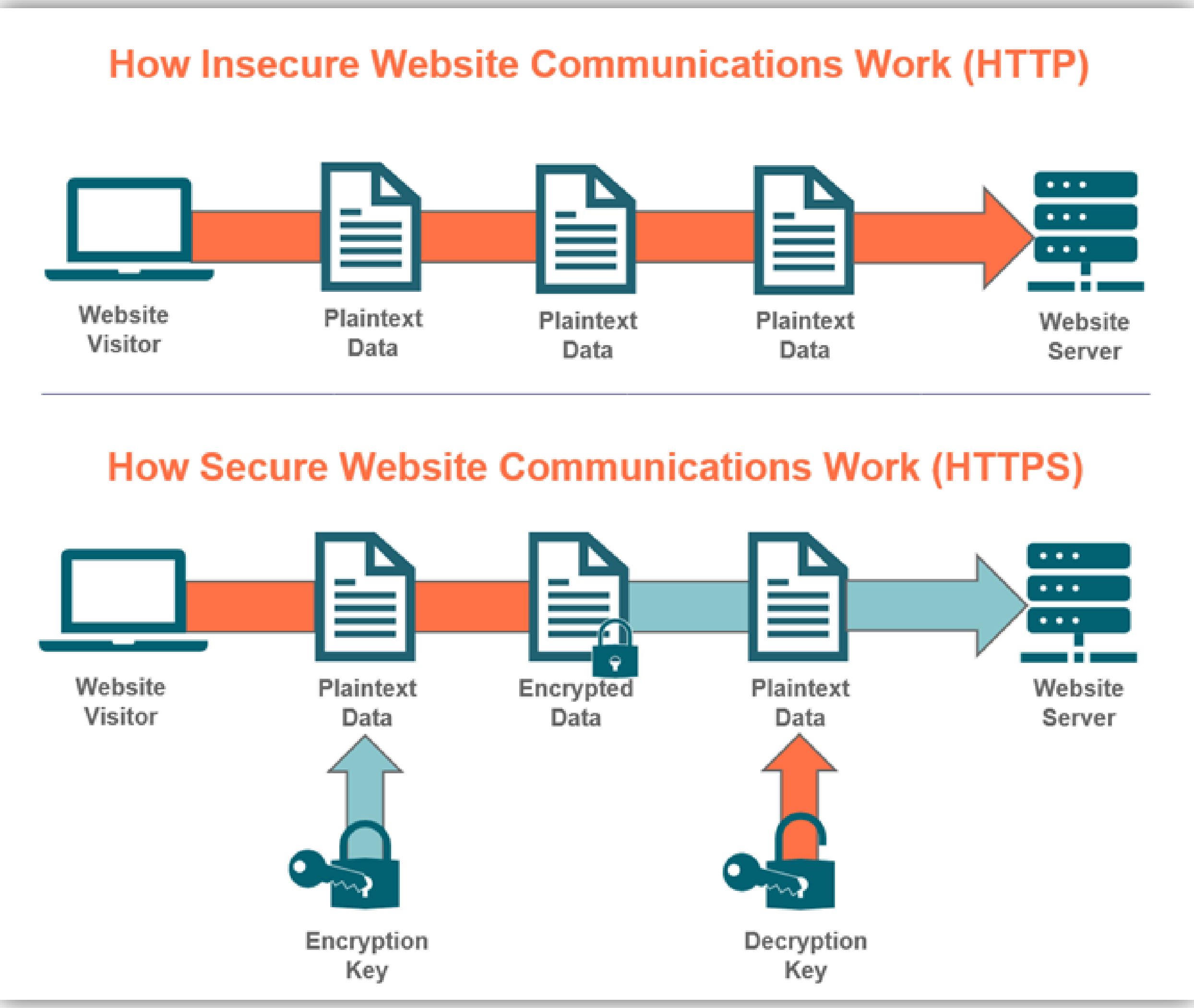
- Encryption
- Hashing
- Digital Signatures
- Public Key Infrastructure (PKI)
- Identity and Access Management (IAM)
- Single Sign On
- Cloud Based Security Groups
- Hardening

Encryption

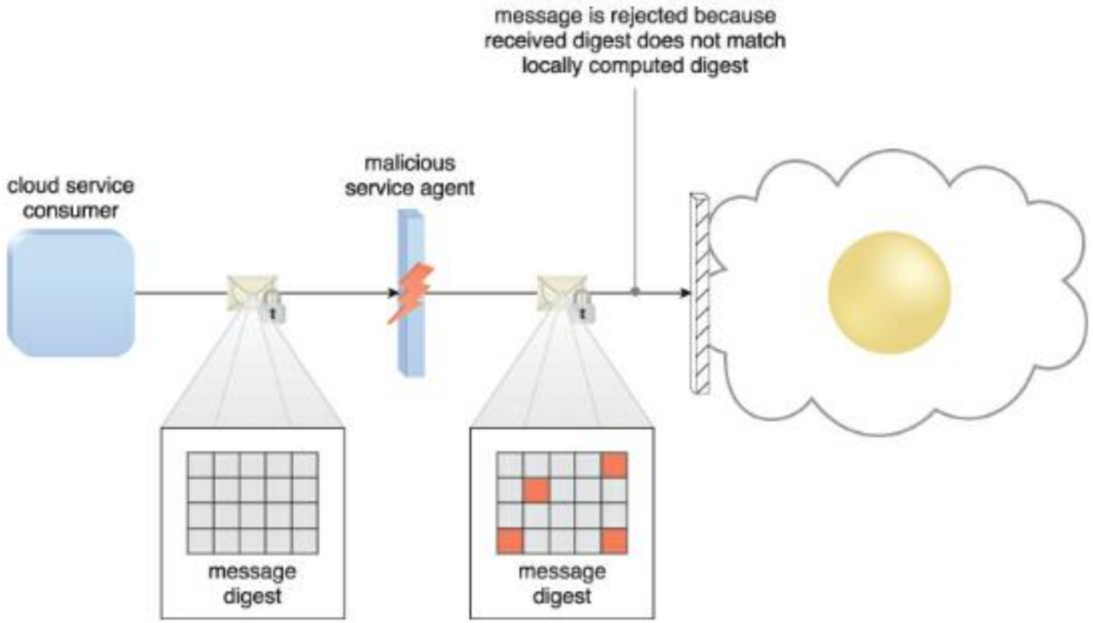

malicious service agent
cloud service consumer
message contents remain confidential

HTTPS

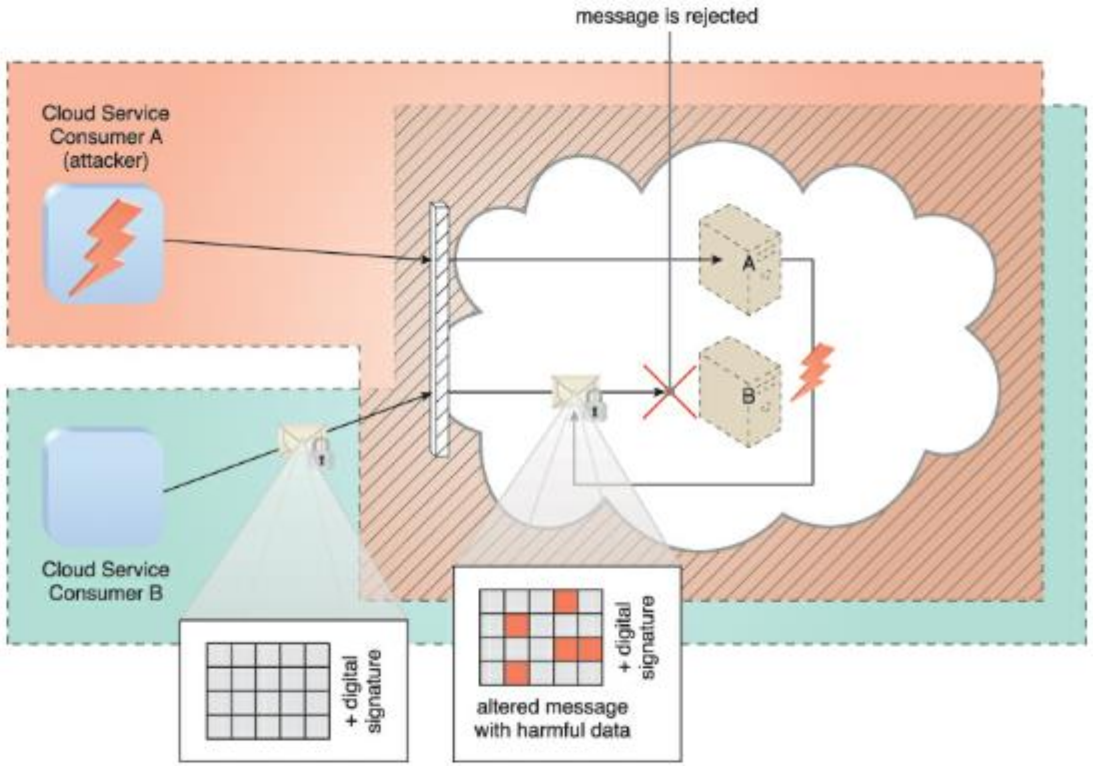
- Ensure message confidentiality through encryption



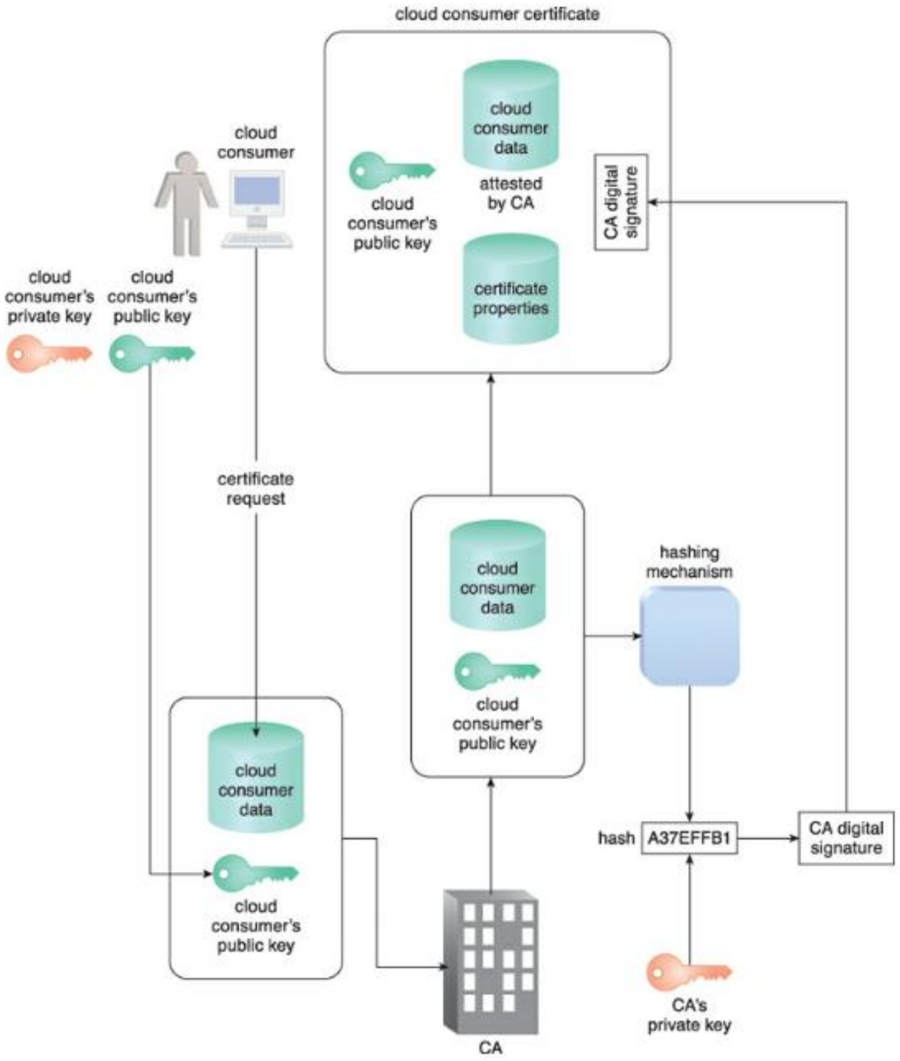
Hashing


message is rejected because received digest does not match locally computed digest
cloud service consumer
malicious service agent
message digest
message digest

Digital Signing



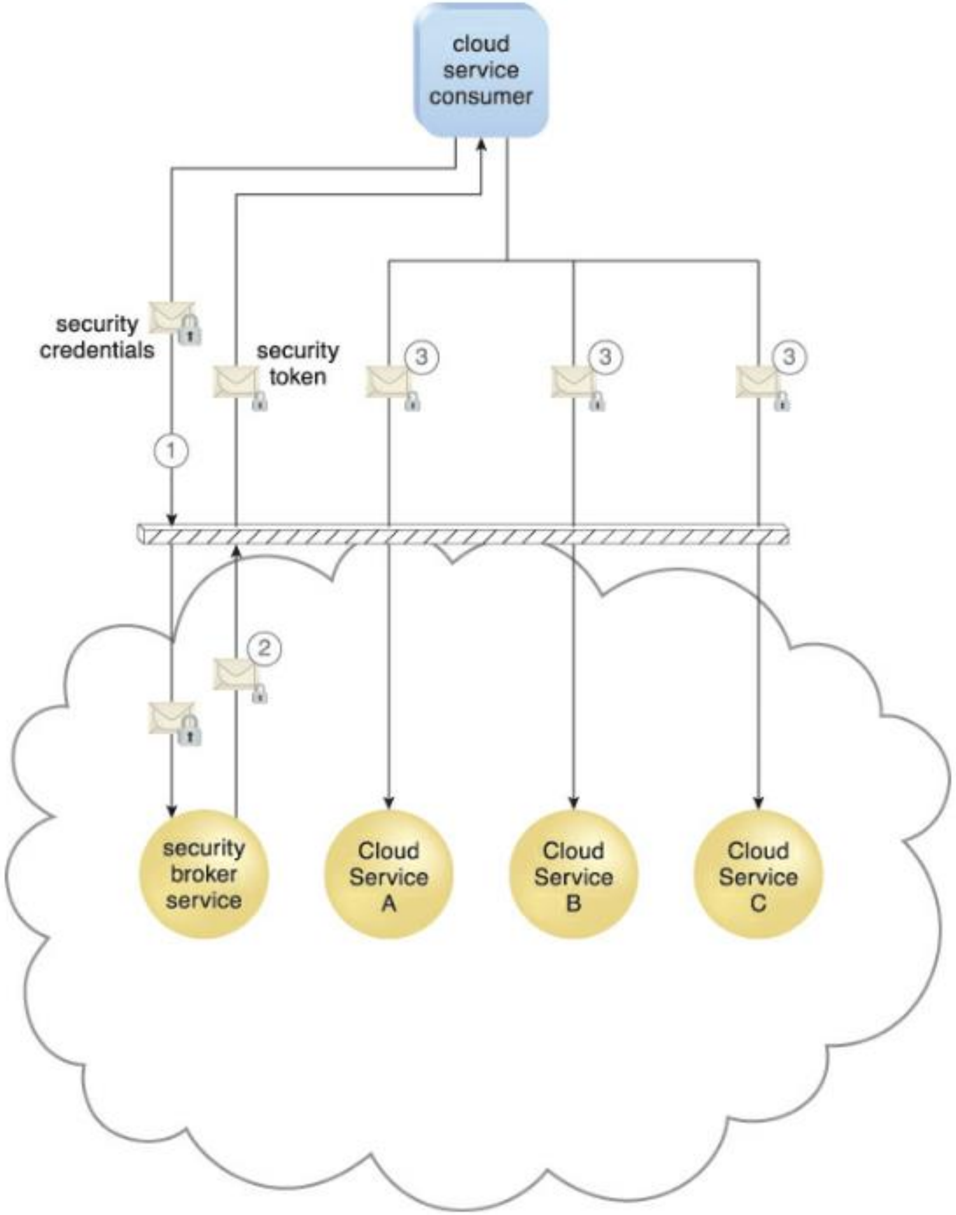
Public Key Infrastructure (PKI)



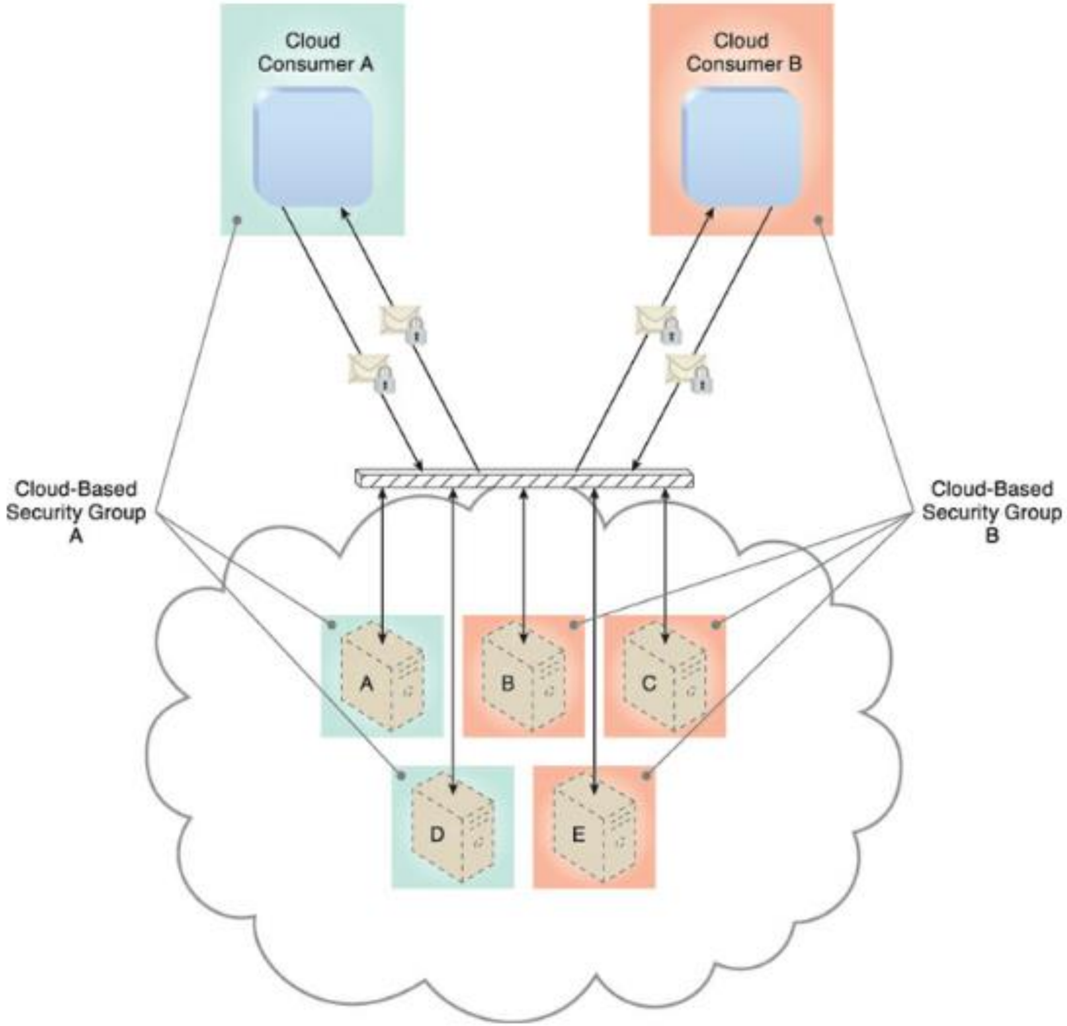
Identity Access Management (IAM)

- Authentication
- Authorisation
- User Management
  - User identities
  - Access groups
  - Managing identities
  - Password policies
  - Role / privilege management
- Credential Management
- Counter insufficient authorization, denial of service, overlapping trust boundaries, virtualization and containerization attack threats.

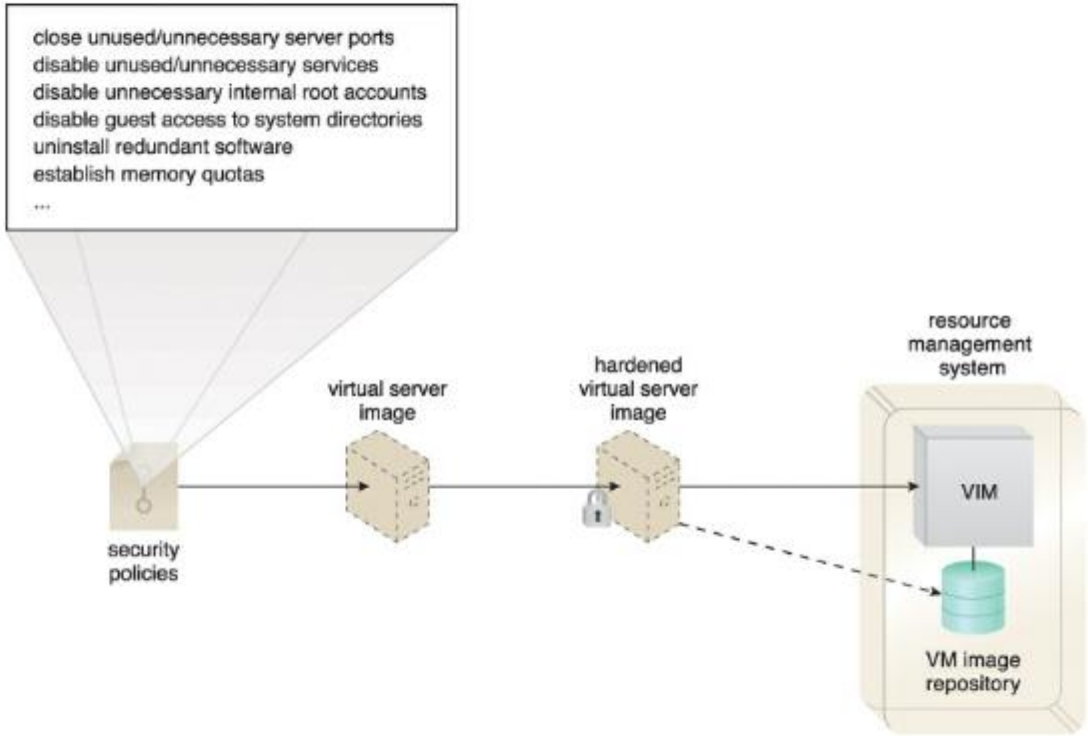
Single Sign On



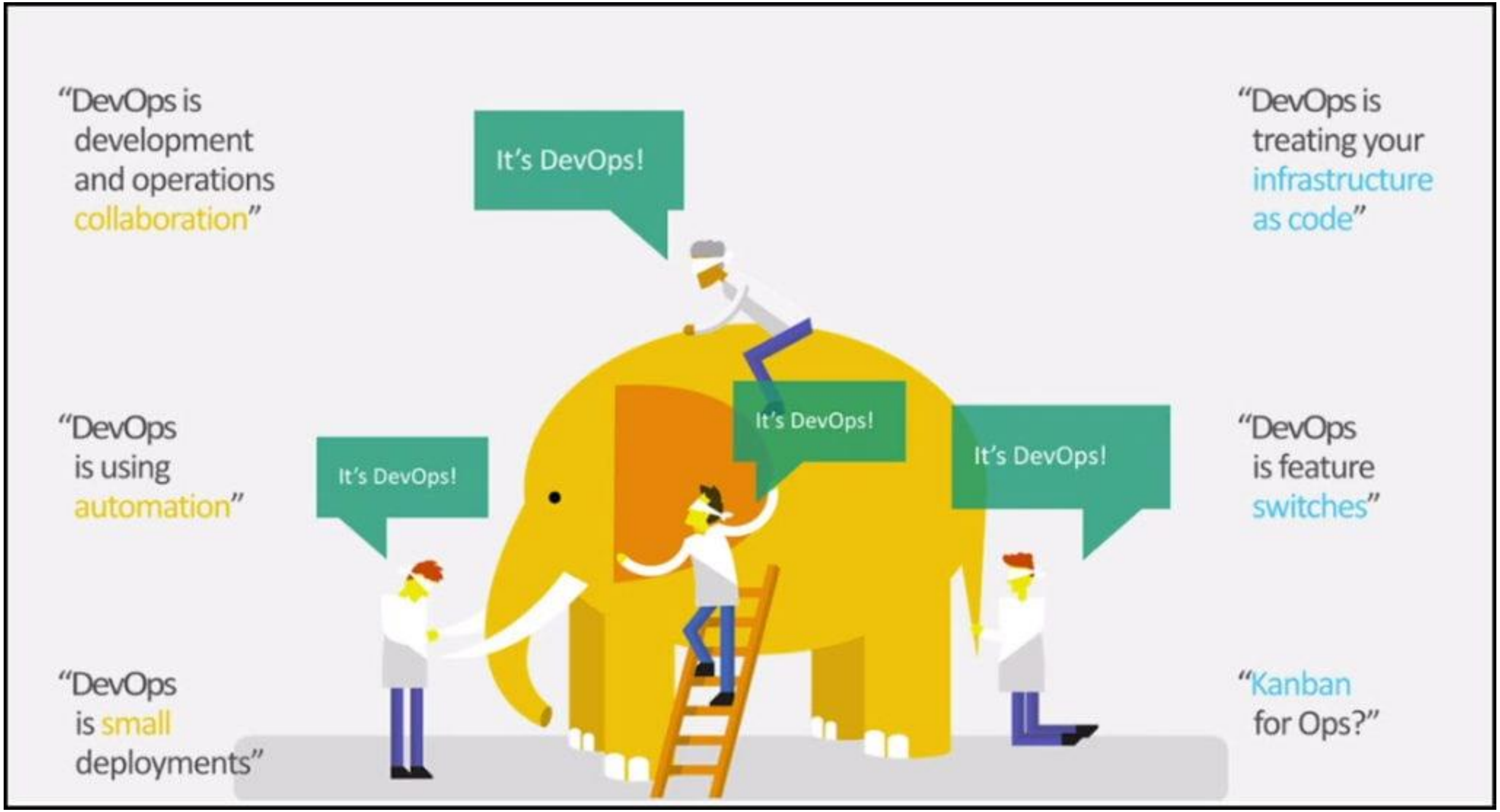
Cloud Based Security Groups



Hardening


close unused/unnecessary server ports
disable unused/unnecessary services
disable unnecessary internal root accounts
disable guest access to system directories
uninstall redundant software
establish memory quotas
...
security policies
virtual server image
hardened virtual server image
resource management system
VIM
VM image repository

What is DevOps



"DevOps is development and operations collaboration"

"DevOps is using automation"

"DevOps is small deployments"

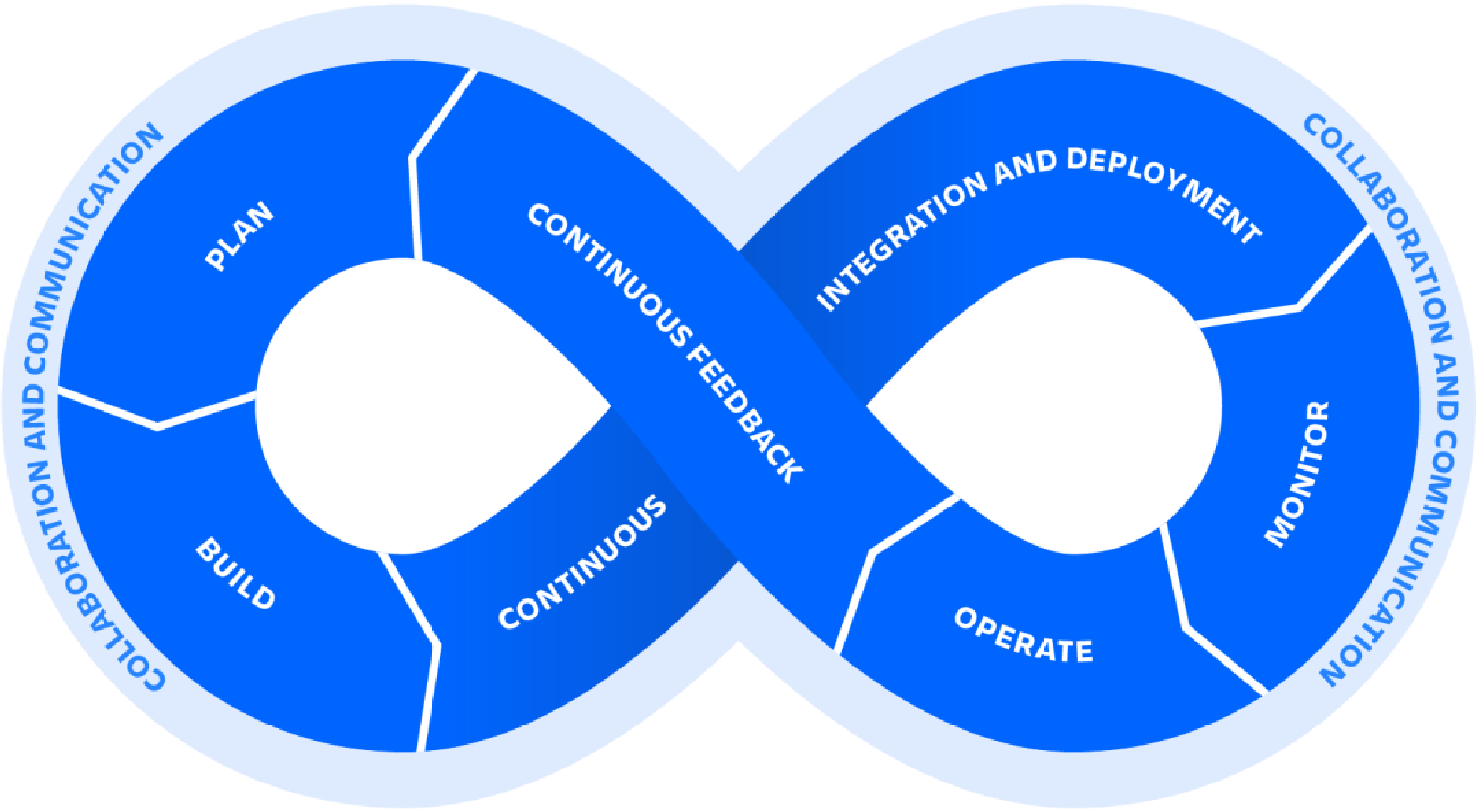
"DevOps is treating your infrastructure as code"

"DevOps is feature switches"

"Kanban for Ops?"

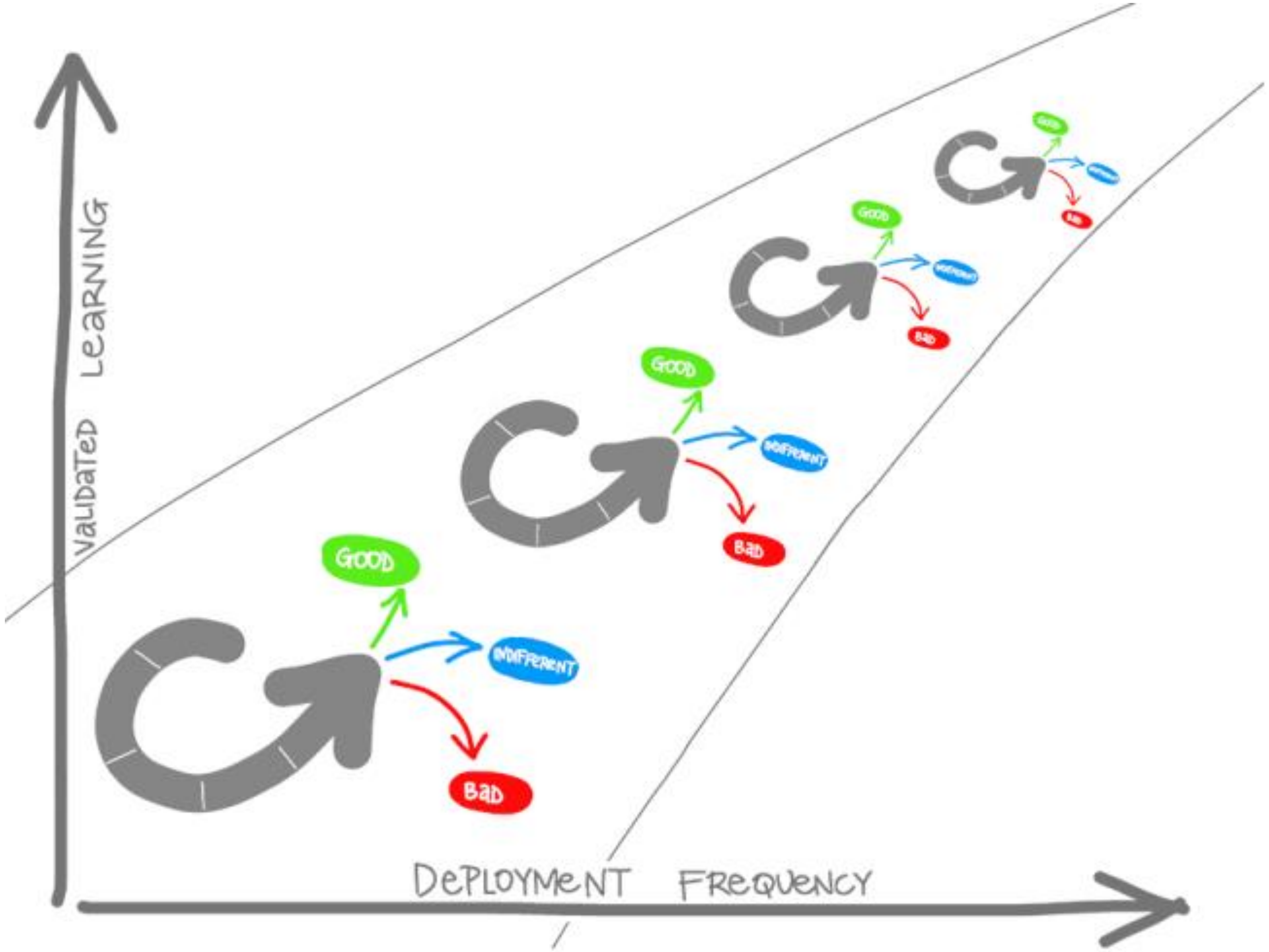
DevOps – A Philosophy

- *DevOps is the union of people, process, and products to enable continuous delivery of value to our end users.*


COLLABORATION AND COMMUNICATION
PLAN
BUILD
CONTINUOUS
CONTINUOUS FEEDBACK
INTEGRATION AND DEPLOYMENT
MONITOR
OPERATE
COLLABORATION AND COMMUNICATION

Decrease the Cycle Time Increase Frequency

- Shorten your cycle time;
  - Smaller batches
  - More automation
  - Hardening release pipeline
  - Improving telemetry
  - Deploy more frequentl



Benefits

Speed

Delivery

Reliability

Scale

Collaboration

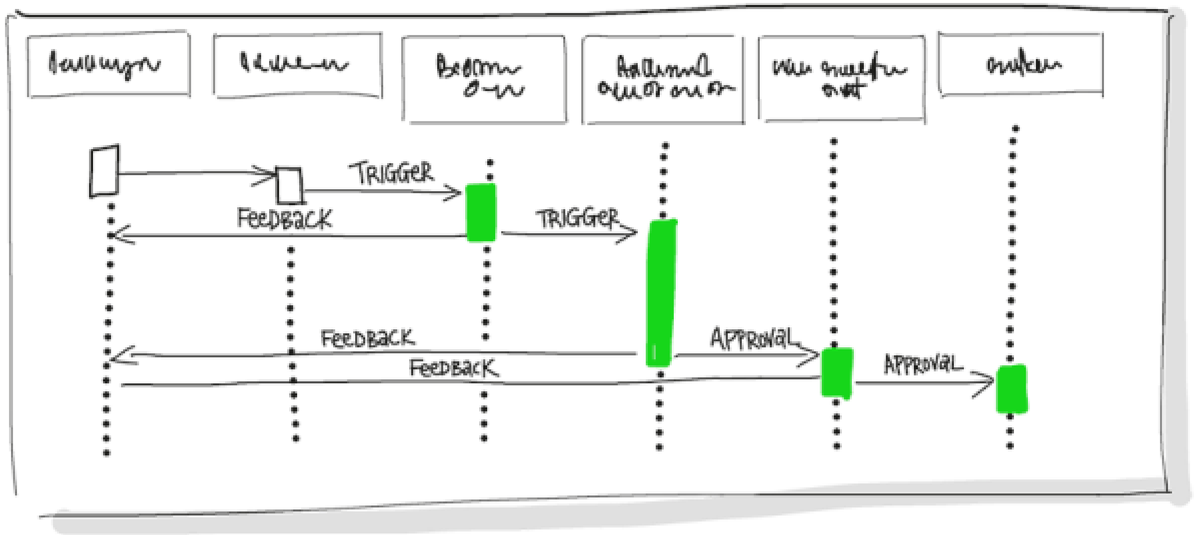
Security

Best Practices

Continuous Integration

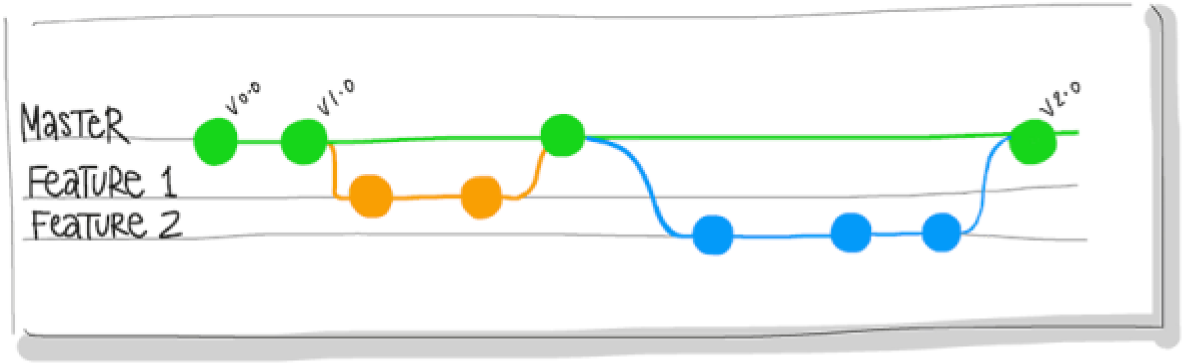
BUILD SUCCEEDED
Completed

Continuous Delivery

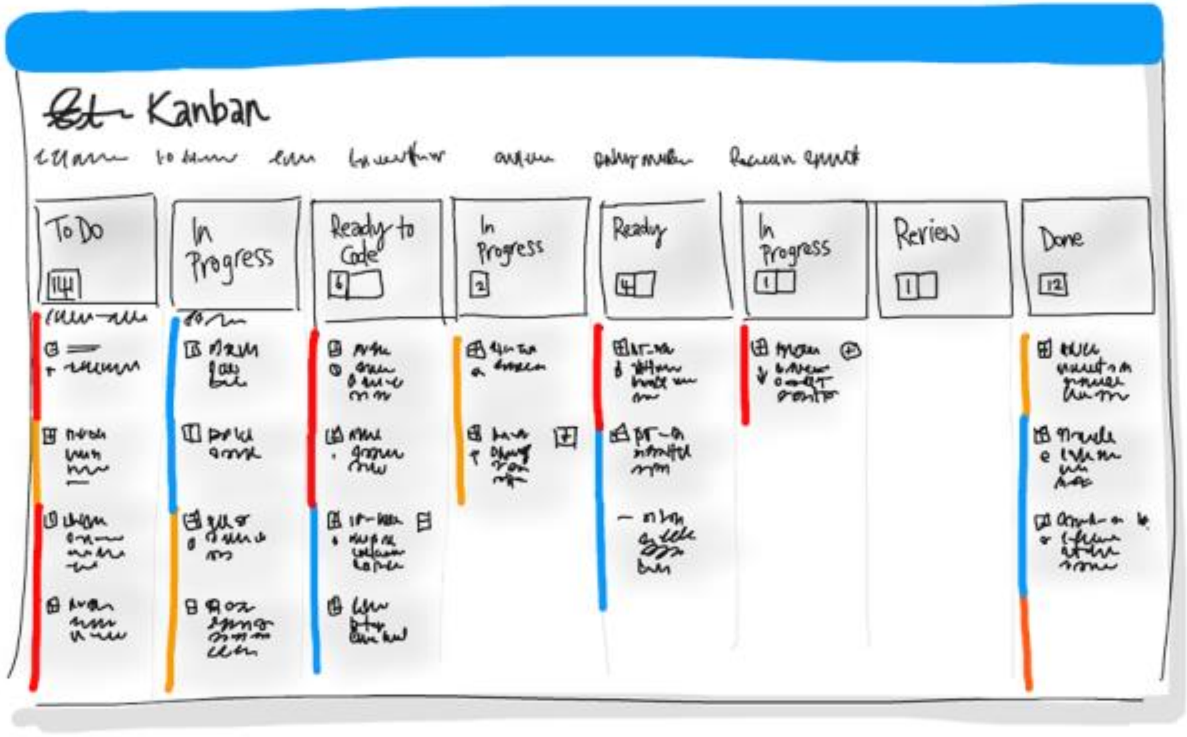


Best Practices

- Version Control

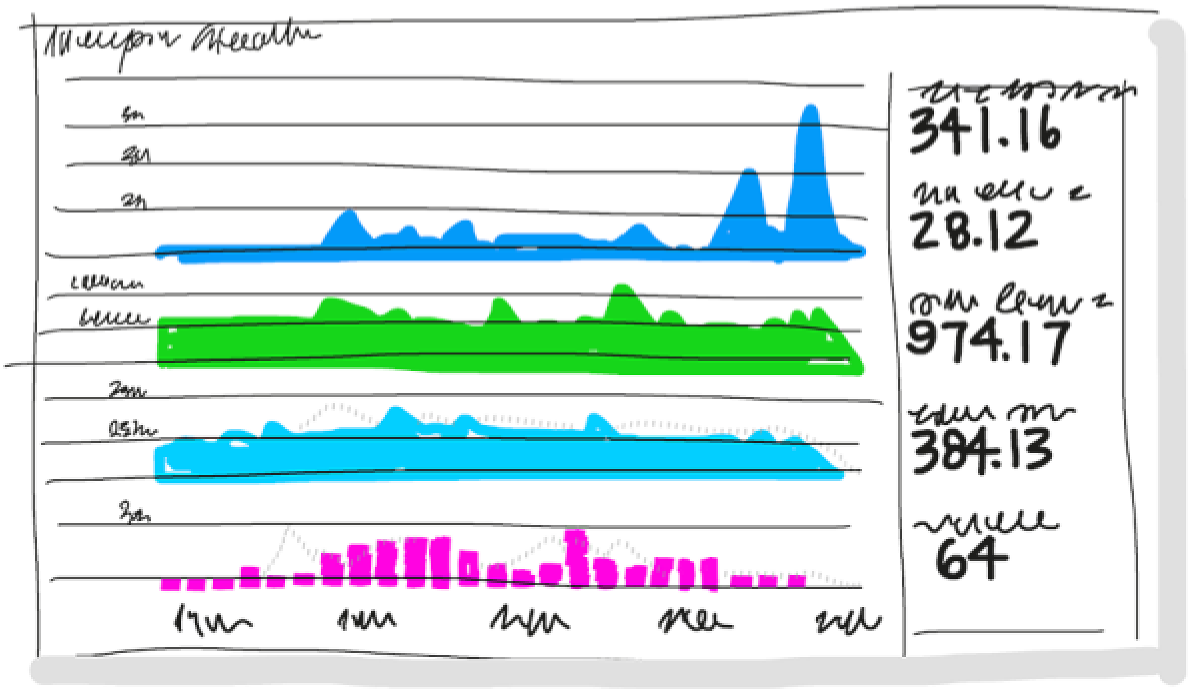


- Agile Planning

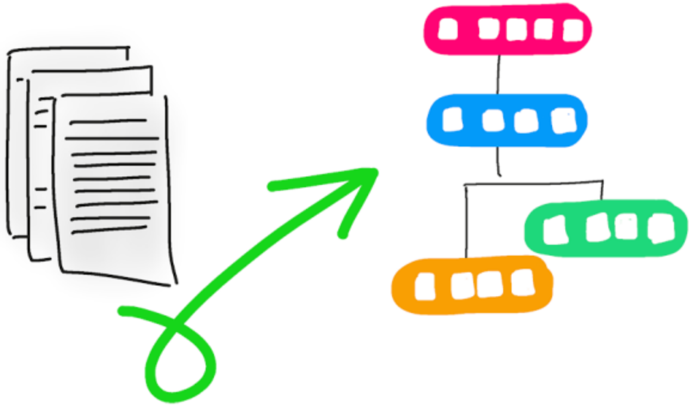


Best Practices

- Monitoring and Logging

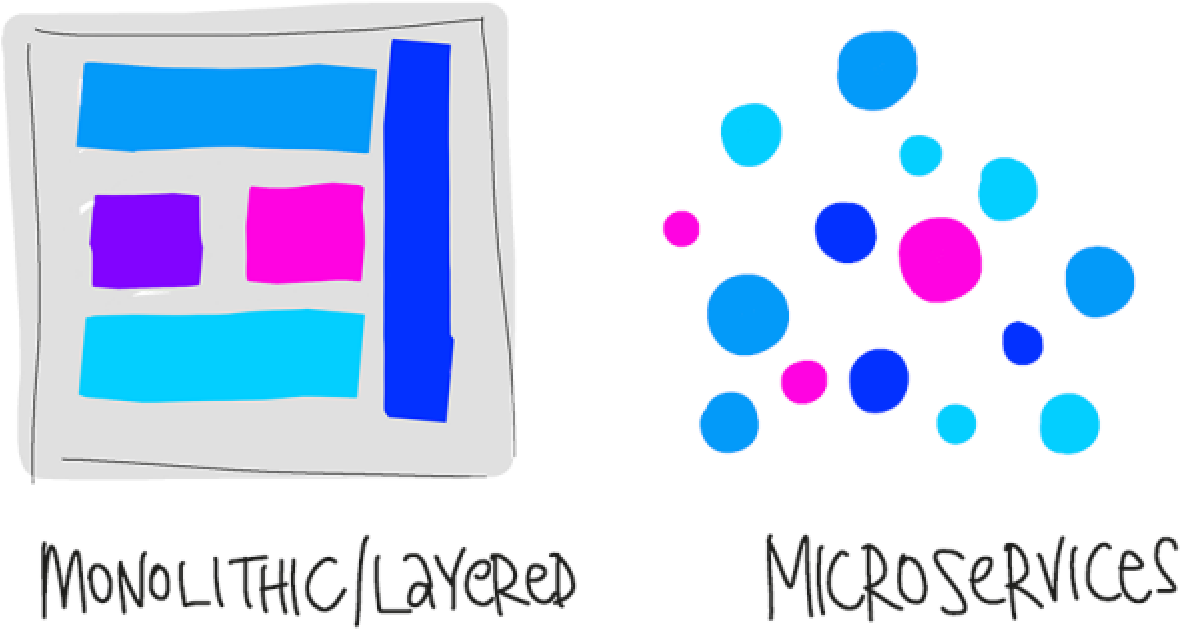


- Infrastructure as Code



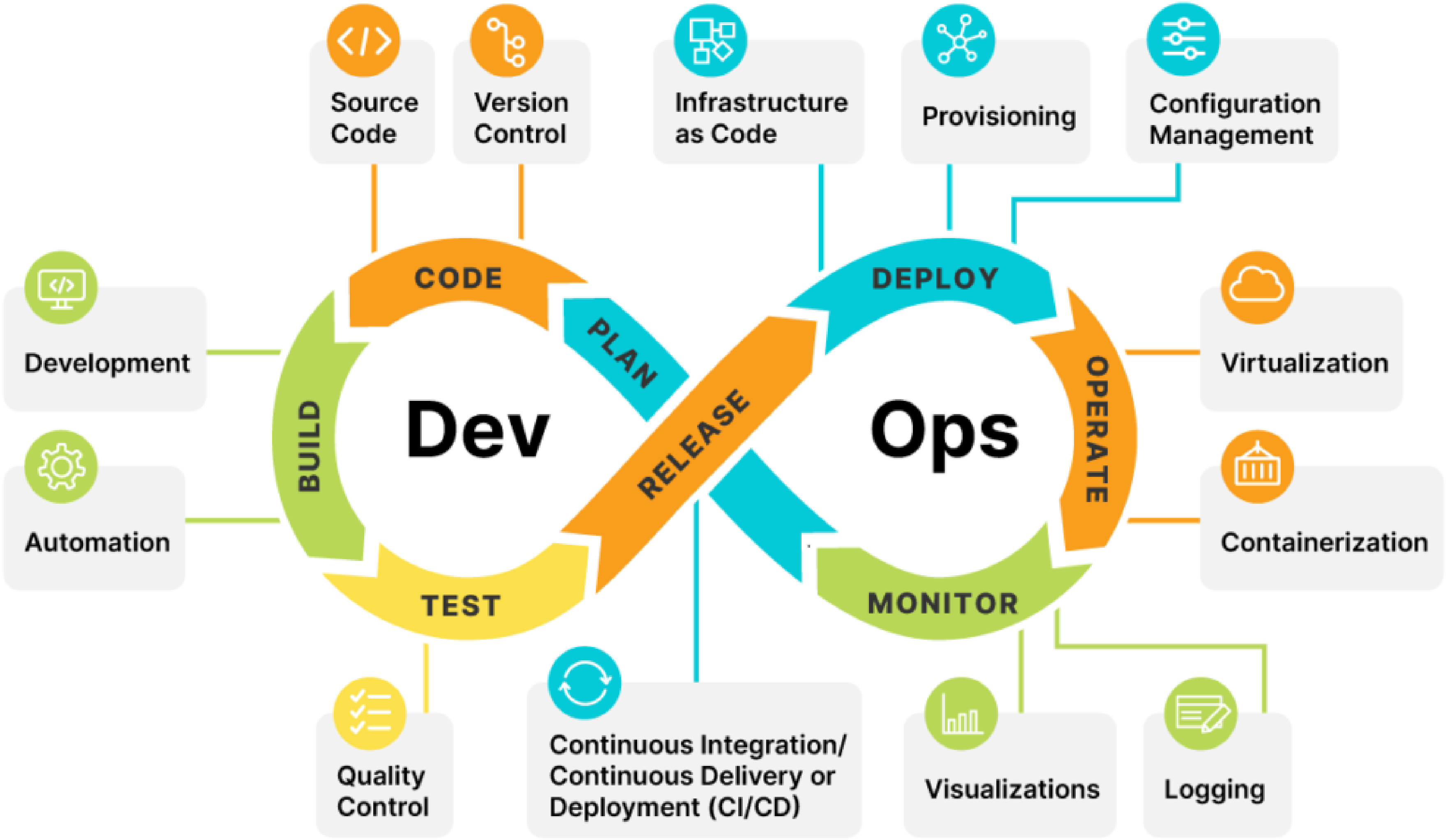
Best Pratices

- Microservices

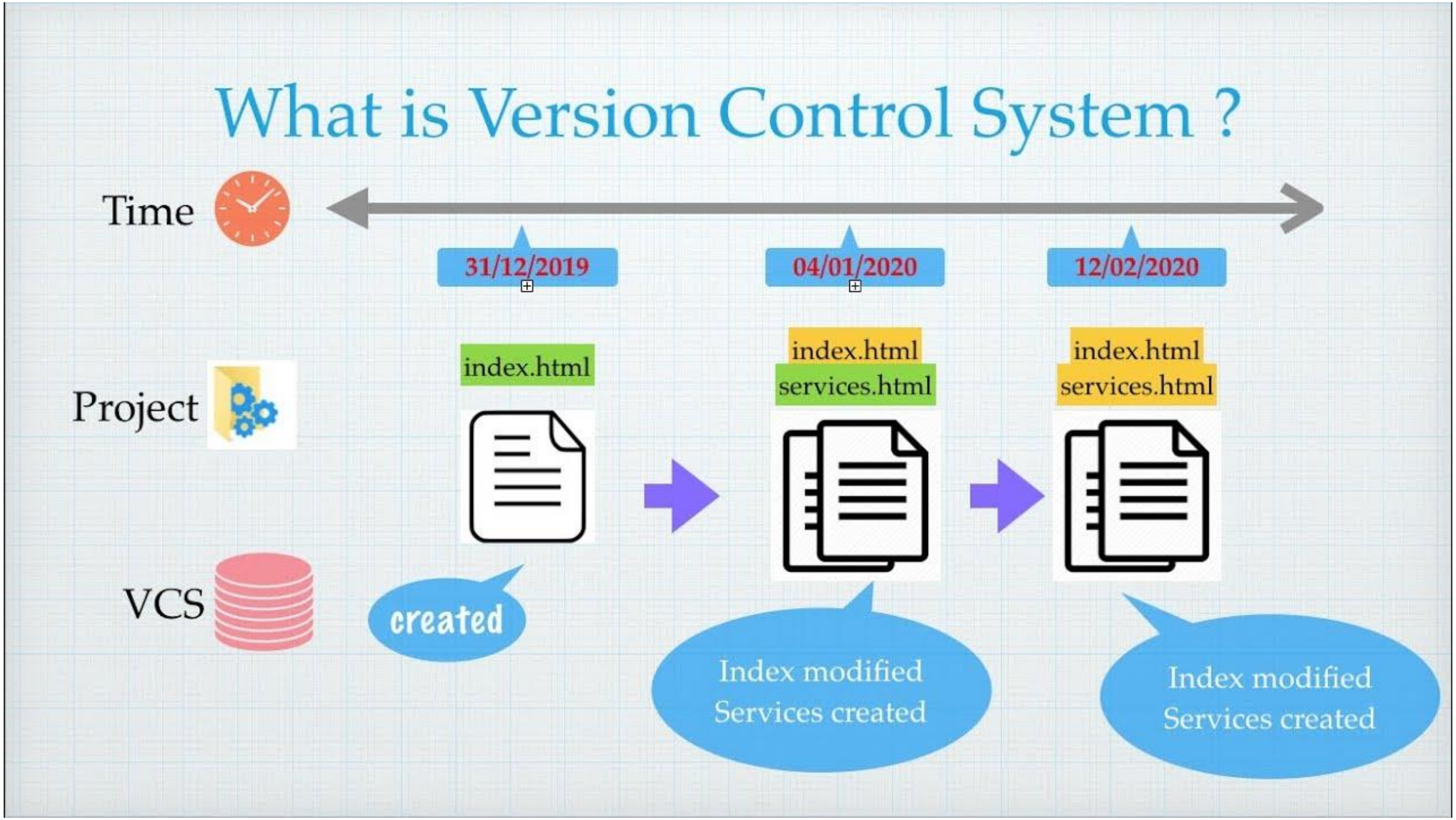


- Communication and Collaboration

DevOps Lifecycle



Why Version Control Systems?



Version Control Systems

SUBVERSION
git
mercurial
TFS
CVS

Benefits of VCS / SCM

History

- Audit trail
- Every change by every individual
- Ability to rollback

Collaboration

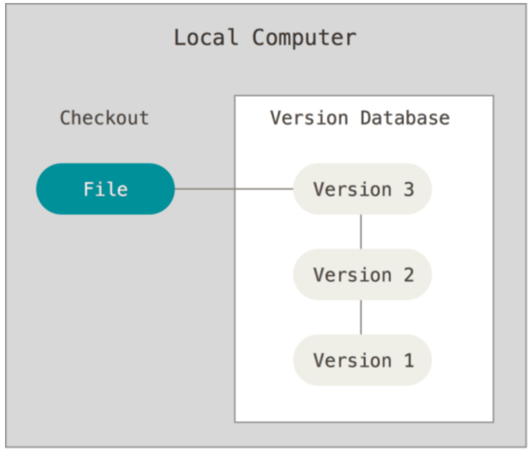
- Branching
- Merging

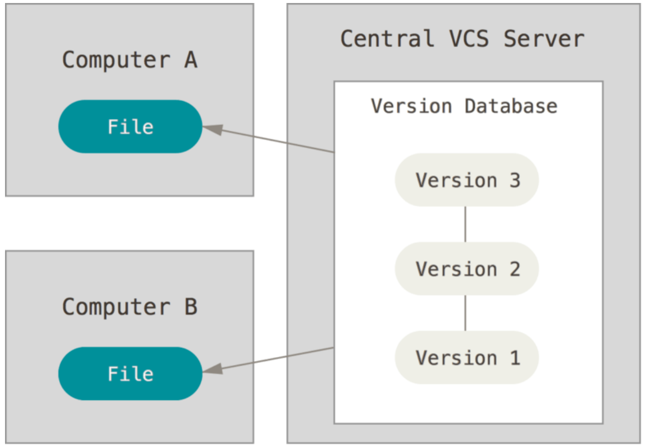
Traceability

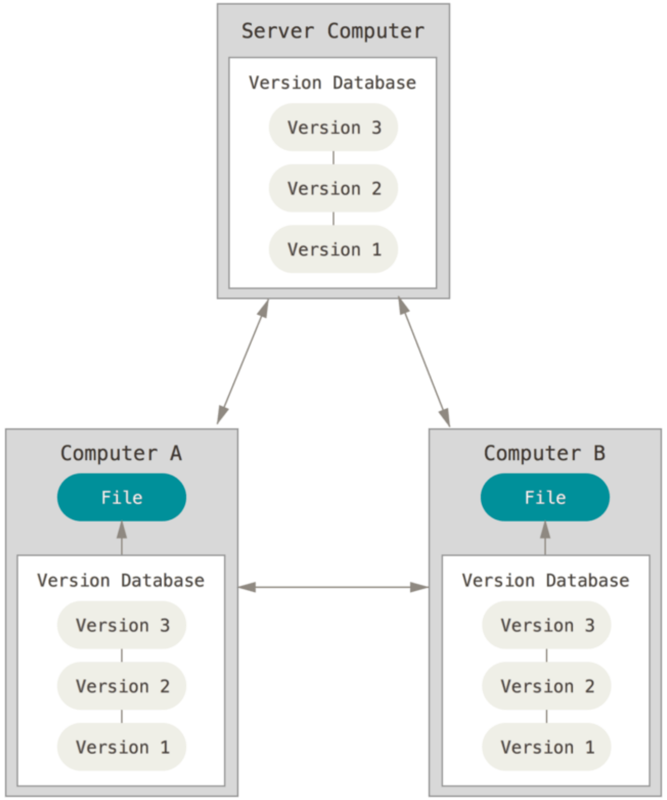
- Grouped changes and commit logs
- Tag to Project Management

Git

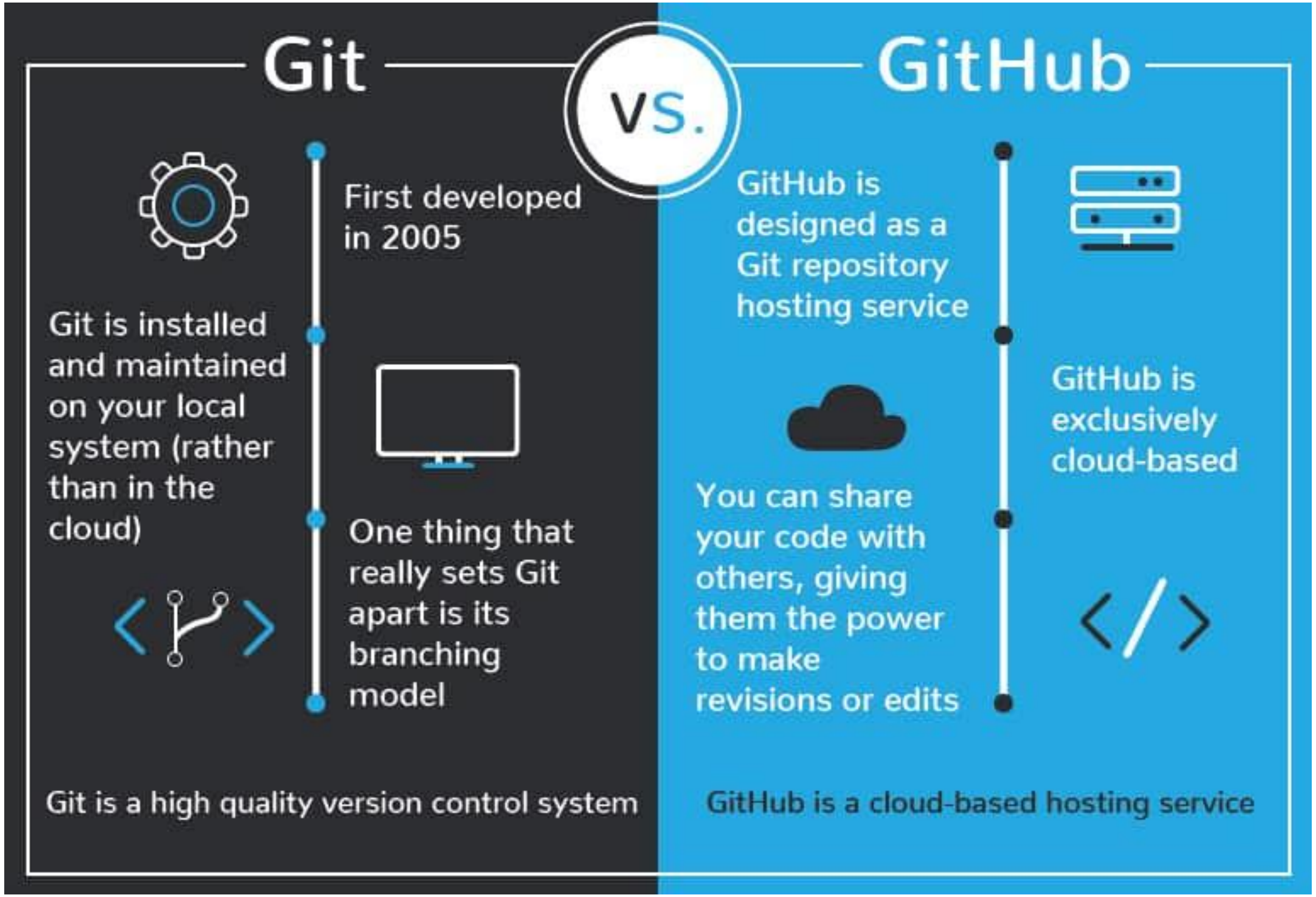
- Used in a large cohort of software development in the world
  - As per Stack Overflow nearly 94% of developers mentioned it as their favourite (2022)
- Developed by Linus Torvalds (main developer of Linux kernel)
  - Because BitKeeper which was hosting the Linux project till then withdrew free use of the product
- DVCS (Distributed Version Control System)







Git vs. Github



Git

First developed in 2005

Git is installed and maintained on your local system (rather than in the cloud)

One thing that really sets Git apart is its branching model

Git is a high quality version control system

GitHub

GitHub is designed as a Git repository hosting service

GitHub is exclusively cloud-based

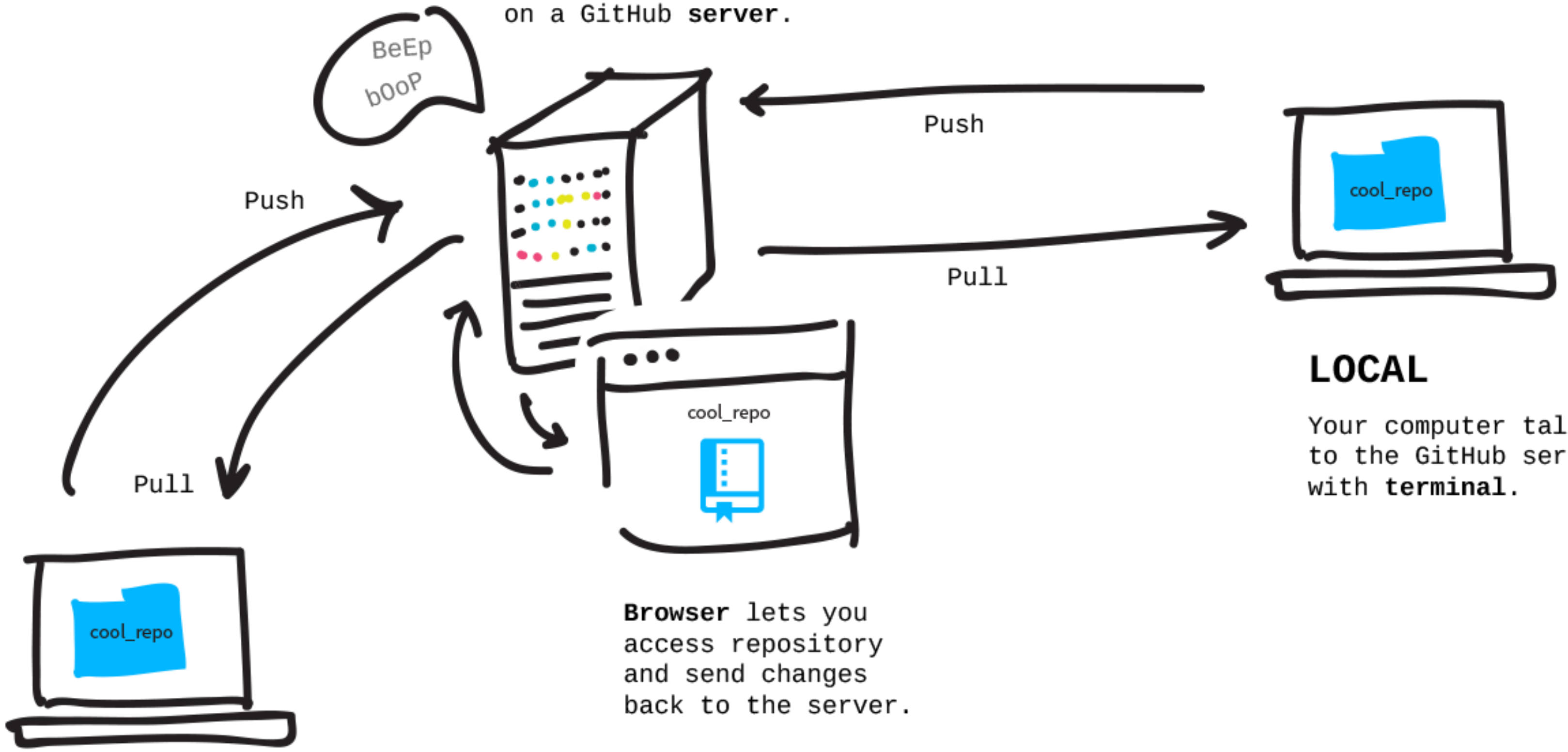
You can share your code with others, giving them the power to make revisions or edits

GitHub is a cloud-based hosting service

Github

REMOTE

Repositories live
on a GitHub **server**.



Browser lets you
access repository
and send changes
back to the server.

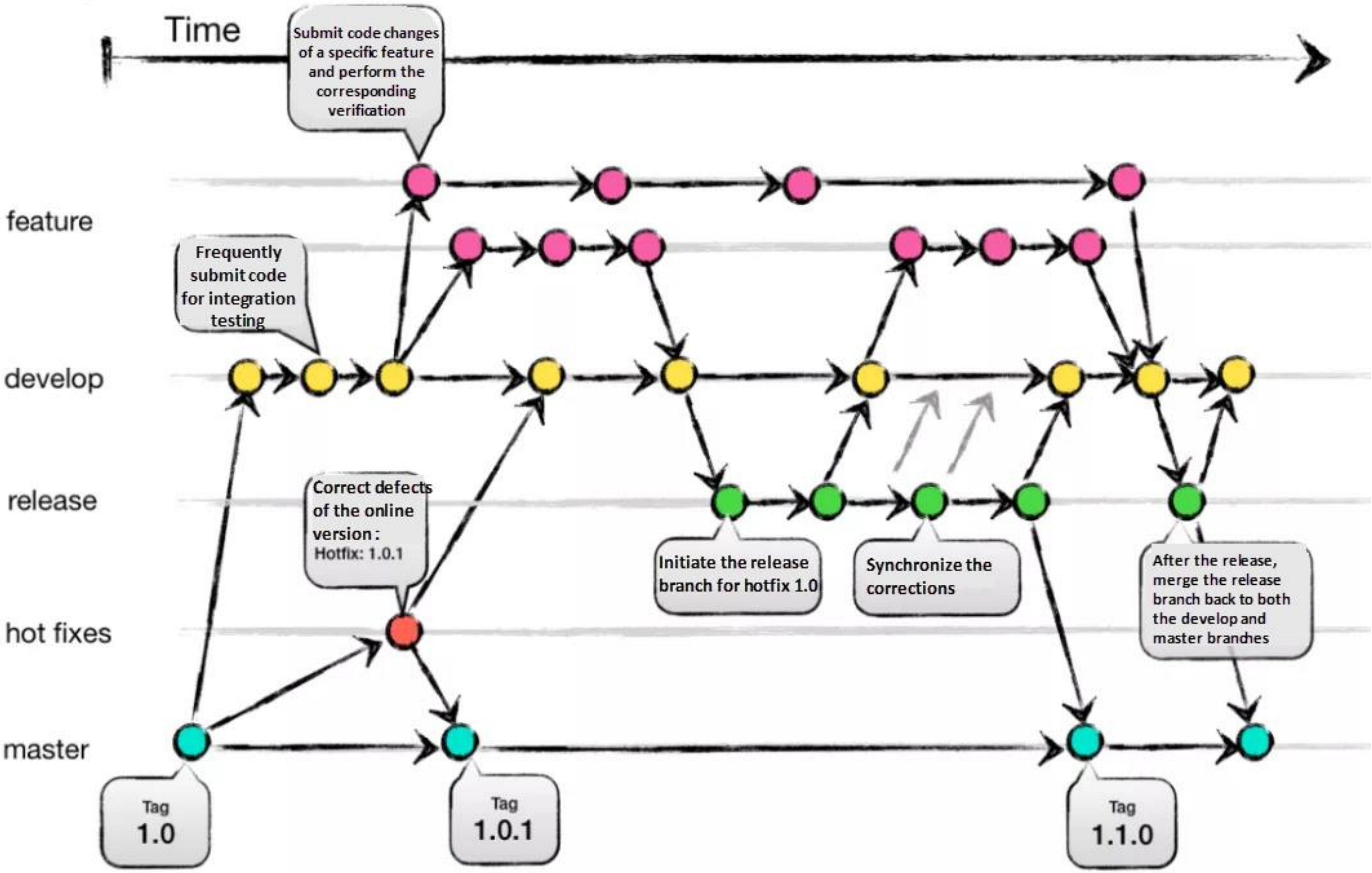
LOCAL

Your computer talks
to the GitHub server
with **terminal**.

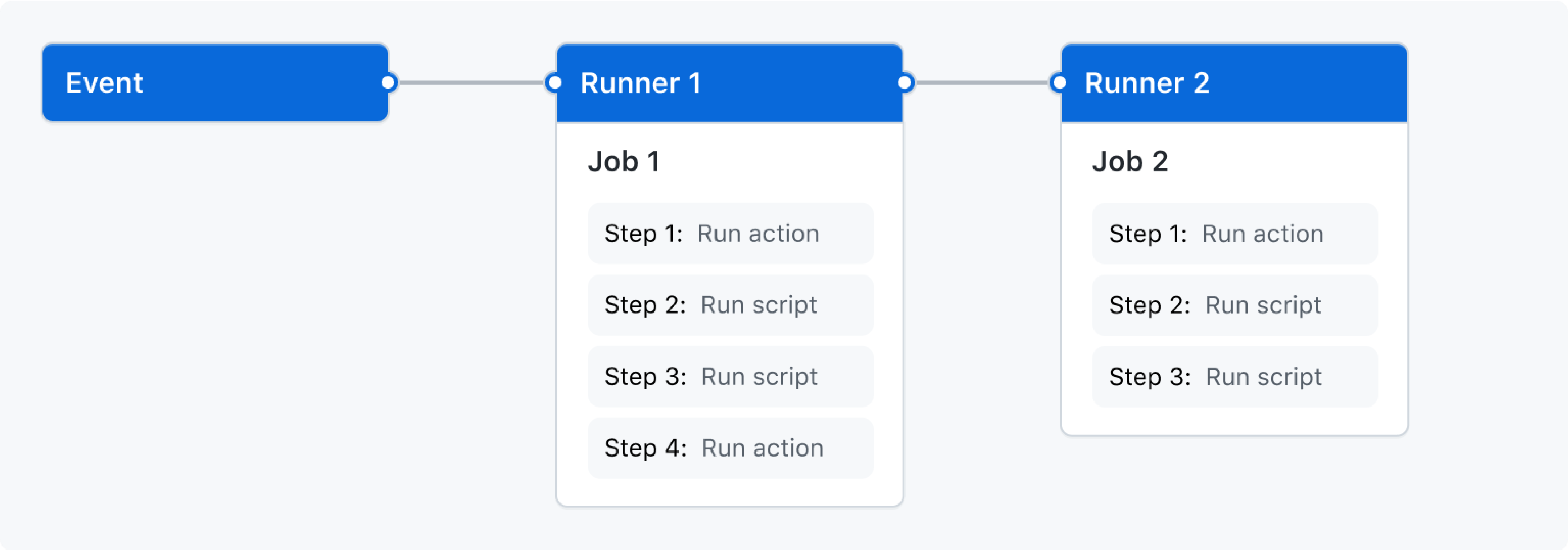
LOCAL

Someone else’s computer
talks to the GitHub server.

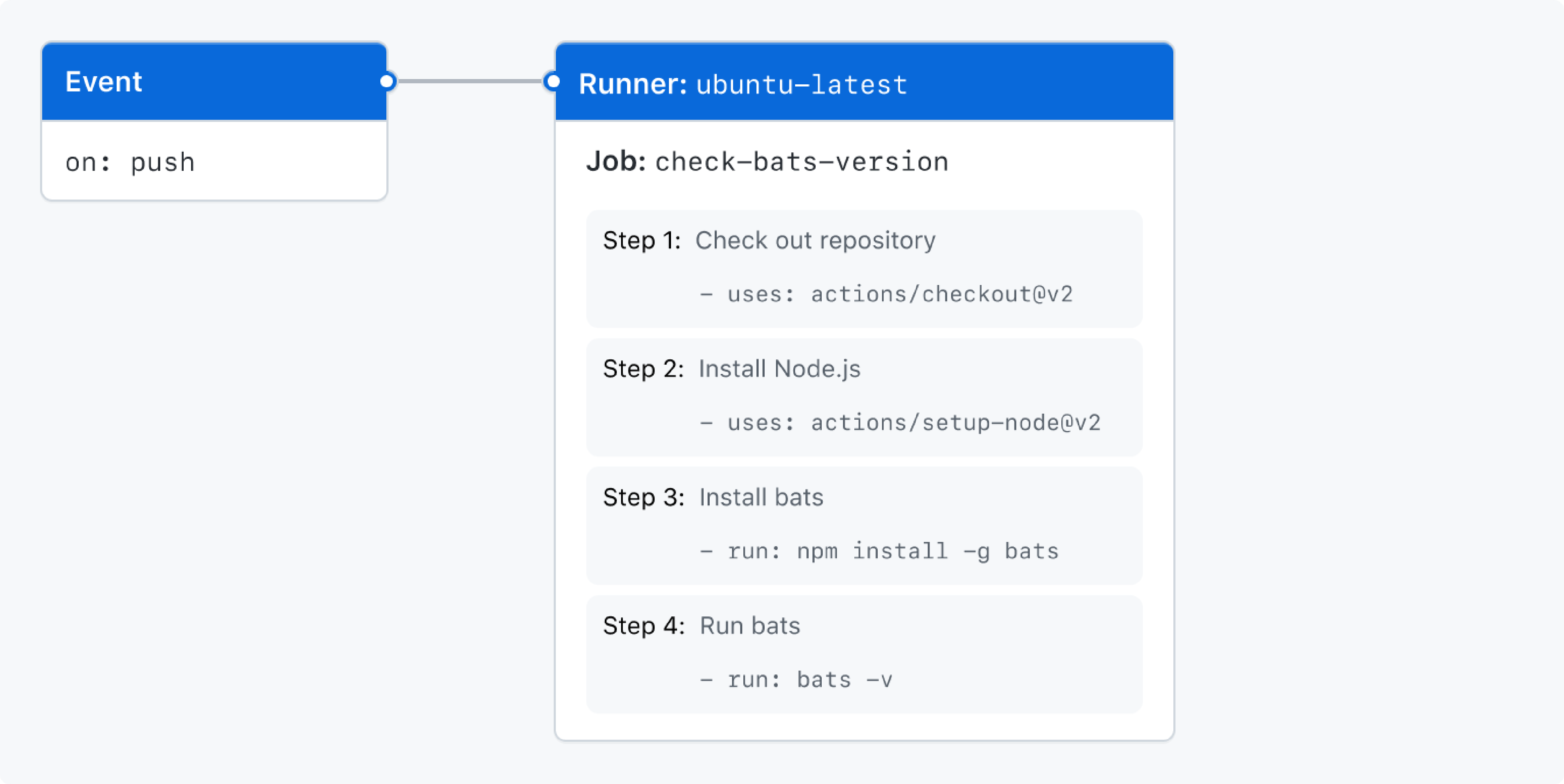
Git Flow


Time
Submit code changes of a specific feature and perform the corresponding verification
feature
Frequently submit code for integration testing
develop
release
Correct defects of the online version : Hotfix: 1.0.1
hot fixes
master
Initiate the release branch for hotfix 1.0
Synchronize the corrections
After the release, merge the release branch back to both the develop and master branches
Tag 1.0
Tag 1.0.1
Tag 1.1.0

Github Actions



Github Actions Runners



Event

`on: push`

Runner: `ubuntu-latest`

Job: `check-bats-version`

Step 1: Check out repository

```
– uses: actions/checkout@v2
```

Step 2: Install Node.js

```
– uses: actions/setup-node@v2
```

Step 3: Install bats

```
– run: npm install -g bats
```

Step 4: Run bats

```
– run: bats -v
```

Exercise – Setup a Github Account

- https://github.com/
- Use an email of your choice (La Trobe student email)
- Create an account

Thank you

LA TROBE UNIVERSITY

Centre for
Data Analytics
and Cognition

http

LinkedIn

Scholar