Banking on the Cloud

Is the cloud secure enough for banking?

Eric Newcomer, Head of Security Architecture and Strategy for Global Consumer Bank (GCB)
Contributions: Ravi Ivaturi, GCB Cloud Security Lead and Alex Shulman, Citi Cloud Security Lead
HPTS: November 5, 2019

Citi / Global Consumer Bank (GCB) Highlights*

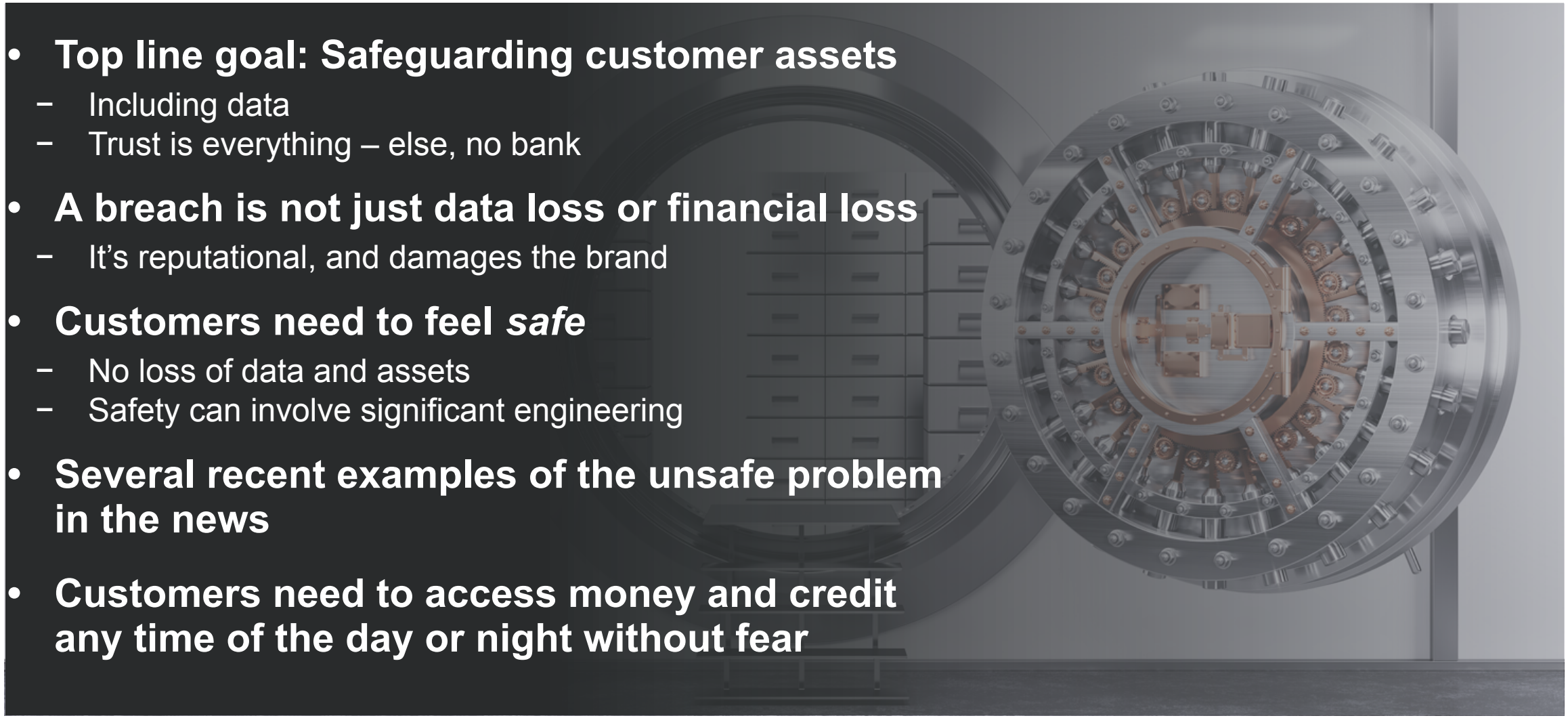
- **Serves more than 110 million clients** in 19 countries.
- **Operates 2,410 branches** and generated $7.6 billion in pretax earnings. business held $307 billion in average deposits, had $423 billion in average assets and included $306 billion in average loans.

Citi Total Revenue: 2018 Annual Report	2018	2017	2016
Global Consumer Banking Net Revenues	**$33.8**	$32.8	$31.6
Institutional Clients Group Net Revenues	**37.0**	36.5	33.9
Corporate/Other Net Revenues	**2.1**	3.1	5.2
Total Net Revenues	**$72.9**	$72.4	$70.8

- **GCB is the world's largest credit card issuer**, Citi is a global leader in payments, with 142 million accounts, $534 million in annual purchase sales, $160 billion in average receivables and premier partners across Citi Branded Cards and Citi Retail Services.
  - Citi Branded Cards provides payment and credit solutions to consumers and small businesses, with 55 million accounts globally. In 2018, Branded Cards generated annual purchase sales of $448 billion and had an average loan portfolio of $112 billion.
  - Citi Retail Services is one of North America's largest providers of private label and co-brand credit cards for retailers, serving 86 million accounts for iconic brands, including Best Buy, ExxonMobil, Macy's, Sears, Shell and The Home Depot. In 2018, Citi Retail Services strengthened its portfolio of leading brands with new partner agreements, strategic renewals and new, industry-leading products. In 2018, Citi Retail Services saw purchase sales of $87 billion, and a loan portfolio ending the year at $53 billion.
- **Citi Commercial Banking** provides global banking capabilities and services to mid-sized, trade-oriented companies within Citi's footprint.

* Citi has two major business divisions: GCB and the Institutional Client Group (ICG). This presentation is mainly from the GCB perspective.
Quotes about GCB are from **About GCB page - https://www.citigroup.com/citi/about/consumer_businesses.html**

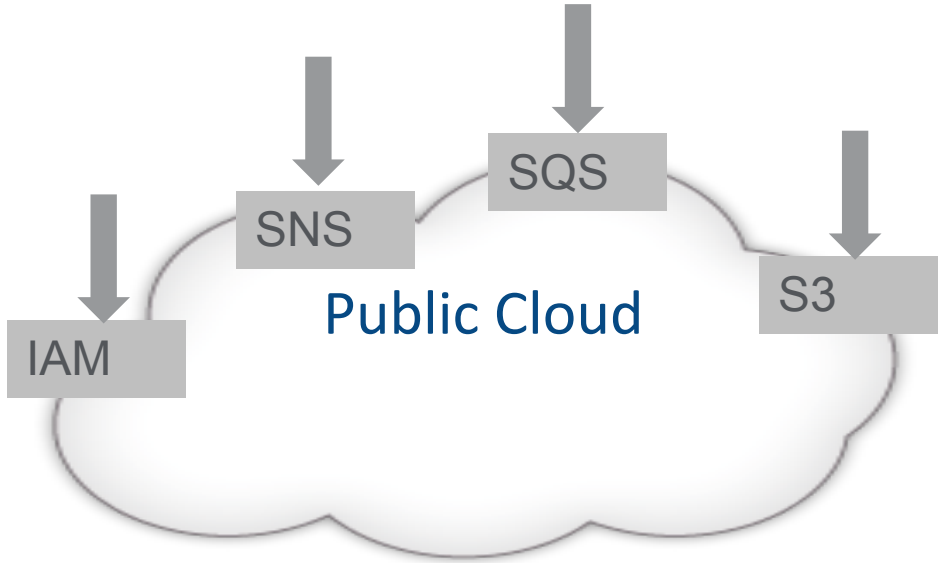
Our customers expect bank grade security



- **Top line goal: Safeguarding customer assets**
  - Including data
  - Trust is everything – else, no bank
- **A breach is not just data loss or financial loss**
  - It's reputational, and damages the brand
- **Customers need to feel *safe***
  - No loss of data and assets
  - Safety can involve significant engineering
- **Several recent examples of the unsafe problem in the news**
- **Customers need to access money and credit any time of the day or night without fear**

Cloud design principles can create risk factors

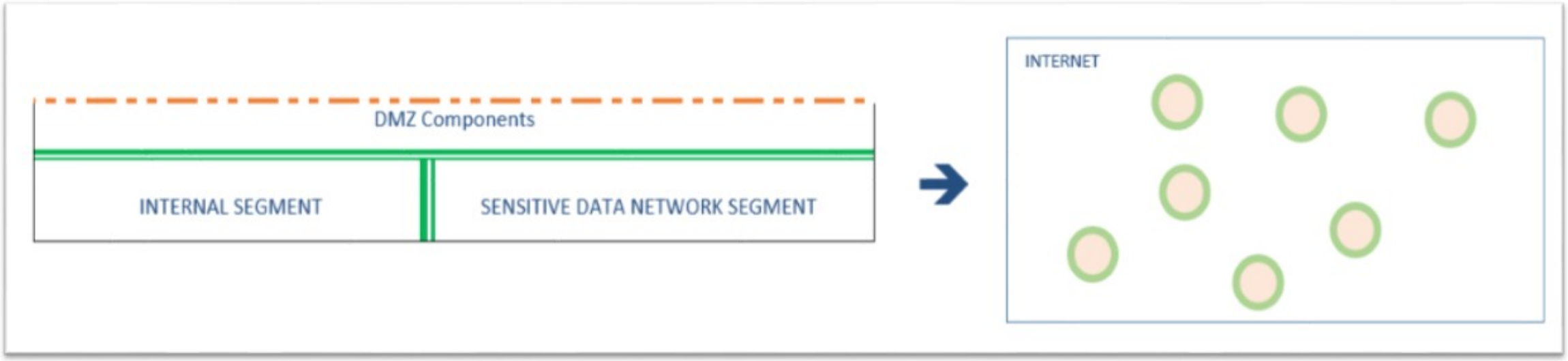
- **Clouds were designed to maximize sharing** (e.g. for on-line shopping) and for Web apps
- **Clouds have differed "perimeter security"** principles defined by:
  - Resource permissions and policies – by design allow internet access
  - IAM systems – by design allow internet access
  - Network constraints - can be bypassed by shared resources
- **Misconfigured policies/permissions** may allow direct external access to company resources (regardless of network and IAM)*
- **Security teams** can not prevent these misconfigurations (since they can be done at the app level)



* See https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html#access_policy-types and https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_aws-services-that-work-with-iam.html

Changing thinking from boxed perimeter to secured components

- **Cybersecurity** was not always well executed even as the traditional boxed perimeter approach
  - Policies and risk management systems are set up to deal with associated vulnerabilities
  - Internal communications and data stores are being encrypted as a next level defense
- **Securing cloud** based applications is more challenging and involves a steeper learning curve
- **Traditional IT organizations** typically try the boxed perimeter approach first
- **A "zero trust"** model is more appropriate for cloud shared infrastructure, networks and perimeters
- **The zero trust model requires** a transition from boxed perimeter to secured components
- **However this adds complexity**, especially in the context of API and config based infrastructure

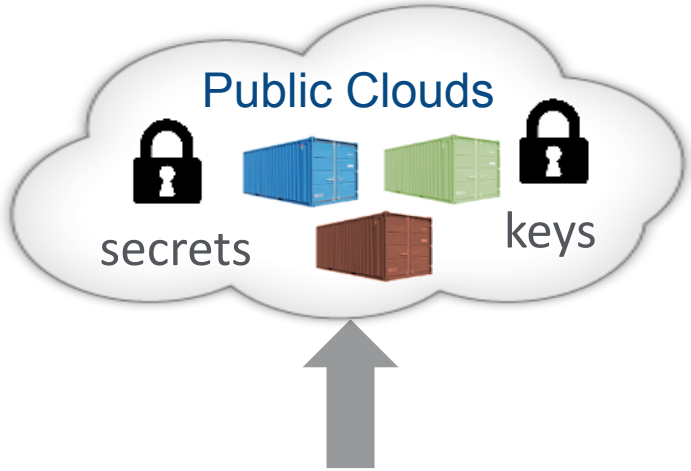


Guardrails are a part of the solution

Guardrails* are code implementing security functionality, e.g.:

- Isolation and protection
- Policy and compliance templates
- Roles and permissions (least priv access)
- Encryption and secrets management
- Event and alerts
- Incident response
- Auto-remediation

- Prevent or detect accidental externally facing services
- Detect and report risky config changes
- Alert the security monitoring center of incidents
- Creates additional complexity in solution designs and governance



Monitor/report/auto correct

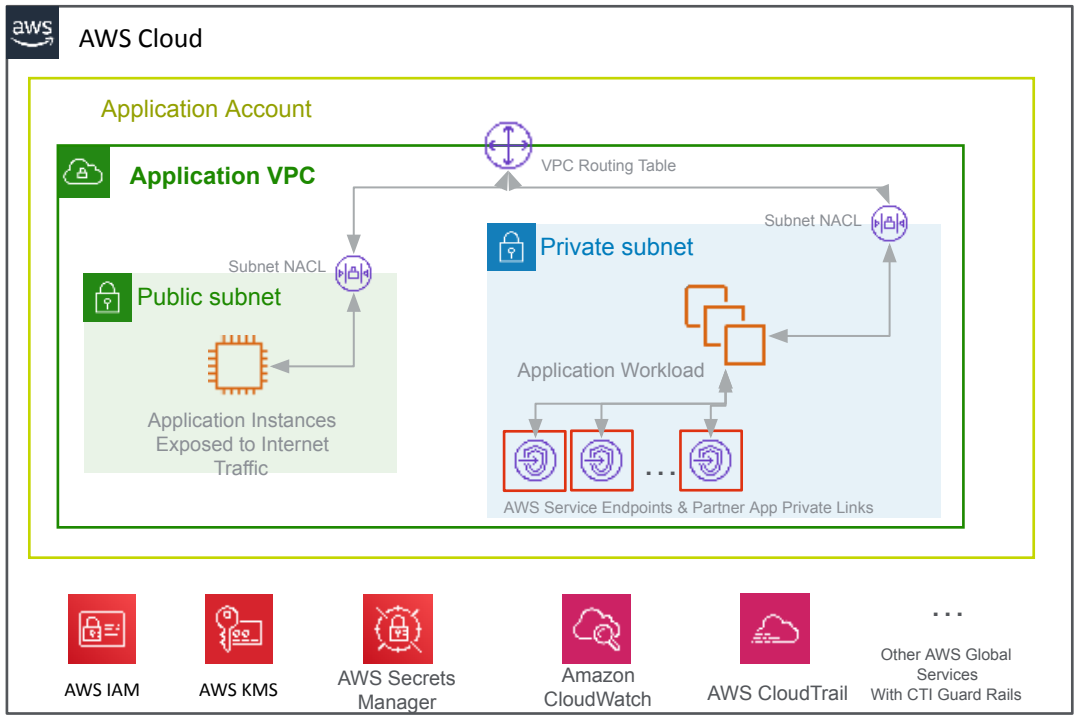
*See: https://devops.com/building-great-cloud-security-guardrails/ and https://securityboulevard.com/2019/05/guardrails-to-innovate-and-stay-secure-in-the-cloud/

Lessons from other bank cloud breaches

- **A recent well known breach was essentially a boxed perimeter failure**
  - A reverse proxy exploit (SSRF Server Side Request Forgery)
  - The components were not secured adequately inside the perimeter
- **Misconfigured WAF was the "open door" that allowed the break in**
  - Attacker took over a privileged account
  - Retrieved the encryption keys for S3 buckets
  - Accessed data for multiple applications
- **Possible failure of governance processes**
  - Development teams often pressured to go to production
  - Just default to all privs etc.
- **Configuration errors are the biggest cause of public cloud incidents (Gartner et al)***
  - Build guardrails
  - Create Cloud Control Matrix for a Cloud Center of Excellence
  - Define application level security patterns
  - Use Cloud Security Posture Management* profiling tools
  - Review access controls, resource policies, identity management coherently and consistently
  - Re-plan patterns to limit the scope of damage

*See https://www.gartner.com/en/documents/3999375/innovation-insight-for-cloud-security-posture-management and https://divvycloud.com/blog/what-is-cloud-security-posture-management/

Example: Application / Workload Isolation (AWS)



AWS Accounts are a critical construct to limit excessive privileges, disable unused services and contain blast radius. Given the Infrastructure-As-Code facet of cloud, this is an essential pattern for cloud security design.

Application Account:

- Account sharing must be avoided at all costs.
- For micro-services, segregation into account must be based on data volume at the minimum, and preferably on a domain model.
- Define a Service Control Policy and enable only utilized services for the application.

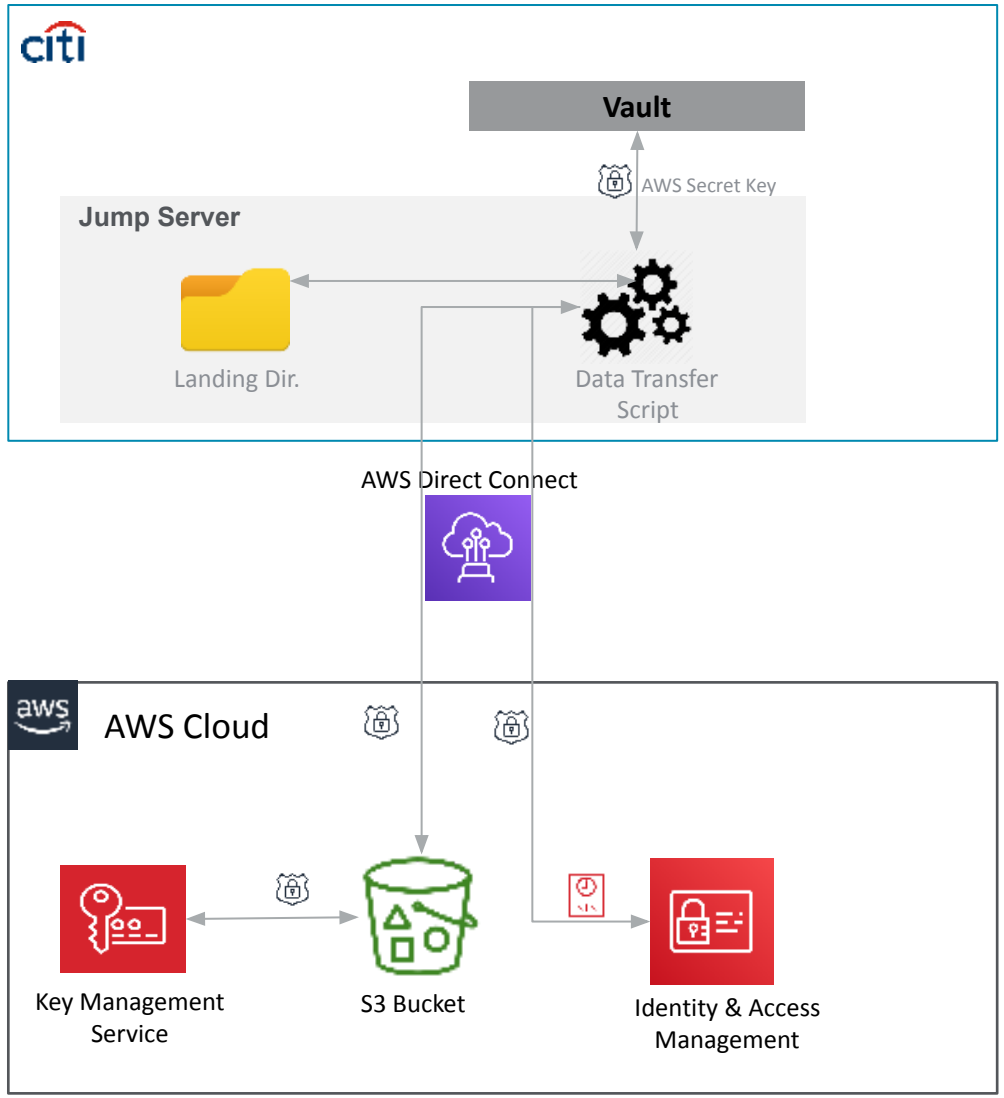
VPC and Routing Table:

- Disable the default allow-all VPC routing table entry.
- Avoid VPC Peering. This potentially exposes other assets in the workload VPC instead of exposing just the required service.

Application Workload & Subnets:

- Deploy the workload in a private subnet. For the workload service/instance, that receives traffic from Internet, place it in a separate / public subnet.
- Define NACLs for each subnet to ensure only expected traffic is allowed.
- Ensure all communication with other applications and services take place via a PrivateEndpoint or a PrivateLink.
- Define a security group for endpoint and permit only necessary access.
- All traffic must use a secure channel such as TLS v1.2 / v1.3.
- For workloads with sensitive data, use-case specific guardrails must be defined – primarily in preventive mode.

Example: Static Content / Data Transfer to AWS S3



Security Pattern for pushing data via script to AWS S3 from On-Prem jump server using AWS Direct Connect.

AWS S3 Controls:

- Dedicated bucket(s) must be used for each use case.
- A Resource Policy for each bucket must be defined to restrict access to data to Principal(s) that require access.
- Whitelist necessary S3 APIs in the resource policy for each principal.
- Define condition(s) in Resource Policy to allow traffic only from the organization's IP addresses used to push data to Direct Connect public interface, or from the Principal Org ID.
- IAM role/principal used by the script to be given only the write access it requires through white-listing..
- Implicit Read access to the S3 objects in the bucket must be explicitly blocked using a DENY read.
- Allow only TLS traffic.

IAM Principal:

- Create and use a dedicated IAM Principal for use by the script.
- Restrict the access using an STS token to desired time. Preferably, half hour or less.
- Define resource-level permission in the IAM role to ensure that the Principal's privileges are limited to the Bucket to which it has to write.

Key Management Service:

- A dedicated key must be created and installed exclusively for the use-case in concern.
- Define a Resource Policy on the Key and limit access to the key to only the Principal used by the script.
- Remove Read-Key privilege to Key Admin and ensure only privileges required for key administration are provided.
- Define an encryption context for the key. This is not secret info and logged in CloudTrail. This will ensure that context around the key that is retrieved and it's purpose/intent is discernable from logs.

On Prem Jump Server:

- Ensure that the AWS Principal's Secret Key is stored in Citi approved password vault
- Ensure landing directory permissions are locked down to ensure only the script has read access and the process landing the files has write access.

Governance/ risk evaluation/ regulation

- **Too much governance** – things get into production anyway (“life will find a way?”)
  - Administrative/management issue
- **How to** keep developers from incurring a significant risk for the firm rather than miss a deadline
  - Especially when they have auto-push capabilities
  - Knowledgeable and skilled developers
- **How to** find the right level of governance
  - Especially for automated, rapid change
  - Security and vulnerability scanning in the pipeline
- **Regulators** are starting to take notice of the cloud
  - Reviewing bank controls for public cloud deployment
  - Of course very concerned about preventing another high impact breach

Questions/Discussion