$ qcc --version

> Checking version information…

Beyond Detection: Deploying a Cloud Response Platform That Actually Responds

- Author: Derek Volmering
- Version: QCC-2025

Press [Enter] to initialize the presentation

```
$ qcc configure --profile 2025
```

> Agenda:

- Intro
- Problem
- Mitigating Control
- Best Practices
- Opensource Options

```
$ qcc auth get-identity --street-creds

> Retrieving background…
    - Recovering IT auditor
    - Passionate cyber defender
    - Head in the clouds
    - Various certs
```

$ qcc auth get-group-info –group ohcr

> curl https://ohcr.ohio.gov/

Authorized in 2019 (Ohio House Bill 52), an organized militia that comprises civilian cyber professionals from academia and private/public enterprise

Core Missions:

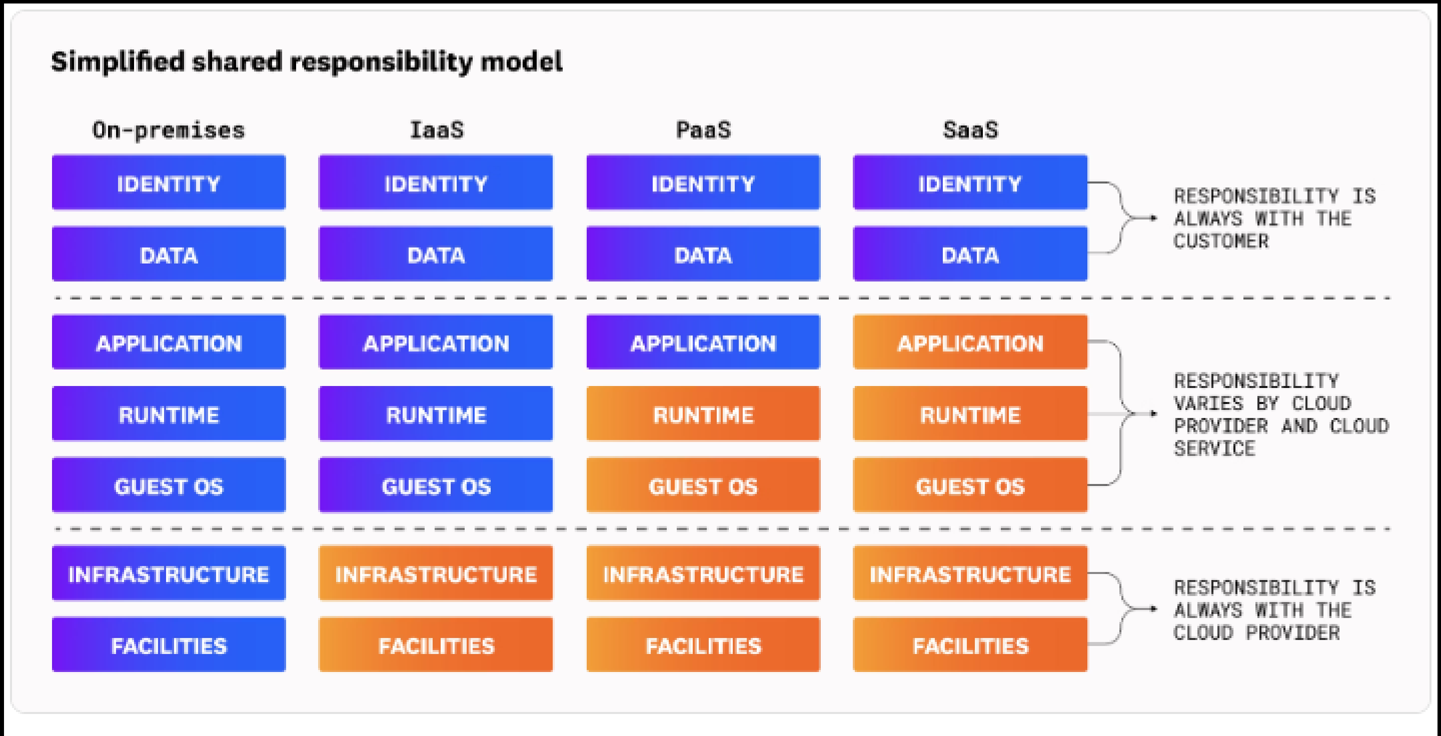
Assist
Educate
Respond

```
$ qcc cloud get-problem

>

    Traditional security tooling
    can’t keep pace with the speed,
    scale, and complexity of the
    cloud.
```

```
$ qcc cloud list-challenges
> loading...
```

Challenges
Increased Attack Surface
Data Overload
Dynamic Infrastructure
Fragmented Security
Evolving Threats

```
$ qcc cloud describe-challenge
--attack-surface

> curl https://www.datadoghq.com/blog/shared-
  responsibility-model/
```

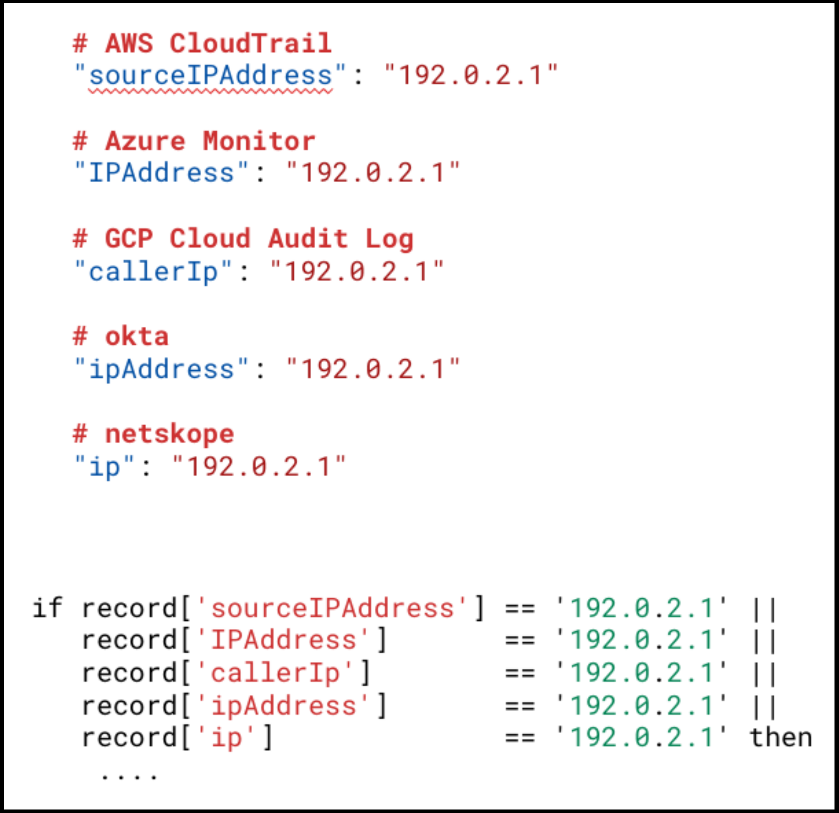


Simplified shared responsibility model

On-premises	IaaS	PaaS	SaaS	
🟪 IDENTITY	🟪 IDENTITY	🟪 IDENTITY	🟪 IDENTITY	RESPONSIBILITY IS ALWAYS WITH THE CUSTOMER
🟪 DATA	🟪 DATA	🟪 DATA	🟪 DATA	
🟪 APPLICATION	🟪 APPLICATION	🟪 APPLICATION	🟧 APPLICATION	RESPONSIBILITY VARIES BY CLOUD PROVIDER AND CLOUD SERVICE
🟪 RUNTIME	🟪 RUNTIME	🟧 RUNTIME	🟧 RUNTIME	
🟪 GUEST OS	🟪 GUEST OS	🟧 GUEST OS	🟧 GUEST OS	
🟪 INFRASTRUCTURE	🟧 INFRASTRUCTURE	🟧 INFRASTRUCTURE	🟧 INFRASTRUCTURE	RESPONSIBILITY IS ALWAYS WITH THE CLOUD PROVIDER
🟪 FACILITIES	🟧 FACILITIES	🟧 FACILITIES	🟧 FACILITIES	

$ qcc cloud describe-challenge --attack-surface --logging

> curl https://redcanary.com/blog/threat-detection/cloud-threat-detection/



```
# AWS CloudTrail
"sourceIPAddress": "192.0.2.1"

# Azure Monitor
"IPAddress": "192.0.2.1"

# GCP Cloud Audit Log
"callerIp": "192.0.2.1"

# okta
"ipAddress": "192.0.2.1"

# netskope
"ip": "192.0.2.1"



if record['sourceIPAddress'] == '192.0.2.1' ||
   record['IPAddress']       == '192.0.2.1' ||
   record['callerIp']        == '192.0.2.1' ||
   record['ipAddress']       == '192.0.2.1' ||
   record['ip']              == '192.0.2.1' then
    ....
```

$ qcc cloud describe-challenge --attack-surface –compare-services

> Calling agentic ai bot “provide comparison of major cloud vendor core features”

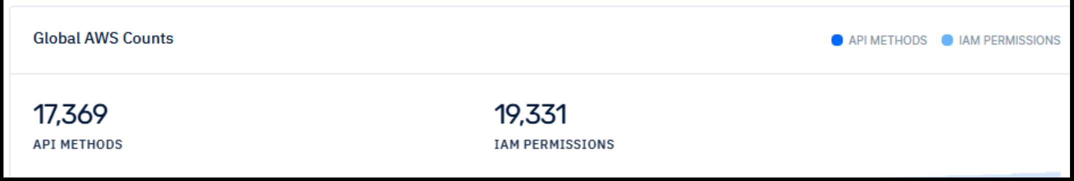
Category / Feature	AWS	Microsoft Azure	Google Cloud (GCP)
Virtual machines / Compute (IaaS)	EC2 (Elastic Compute Cloud)	Virtual Machines (VMs)	Compute Engine
Managed Kubernetes	EKS (Elastic Kubernetes Service)	AKS (Azure Kubernetes Service)	GKE (Google Kubernetes Engine)
Serverless functions (FaaS)	Lambda	Azure Functions	Cloud Functions
Serverless containers / run containers (serverless PaaS)	Fargate / App Runner	Azure Container Instances / App Service	Cloud Run
Object storage	S3 (Simple Storage Service)	Blob Storage (Storage Accounts)	Cloud Storage
Block storage	EBS (Elastic Block Store)	Managed Disks	Persistent Disk
Managed file / NFS storage	EFS (Elastic File System) / FSx	Azure Files / Azure NetApp Files	Filestore
Relational managed DB (PaaS)	RDS (MySQL/Postgres/SQL Server), Aurora	Azure Database for MySQL/Postgres / Azure SQL / Managed Instance	Cloud SQL
Analytical data warehouse	Redshift	Synapse Analytics	BigQuery
NoSQL / key-value / document DB	DynamoDB	Cosmos DB (multi-model)	Firestore / Bigtable (wide-column)
Message queue / simple queue	SQS	Storage queue / Service Bus (queue)	Cloud Tasks / Pub/Sub (for messaging)
Pub/sub / streaming	Kinesis (Data Streams) / MSK (Kafka)	Event Hubs / Service Bus / Event Grid	Pub/Sub / Dataflow (stream processing)
API gateway / management	API Gateway / API Gateway (HTTP APIs)	API Management	API Gateway / Endpoints
DNS	Route 53	Azure DNS	Cloud DNS
CDN (Content Delivery Network)	CloudFront	Azure CDN	Cloud CDN
Monitoring / logging / observability	CloudWatch / X-Ray	Azure Monitor / Application Insights	Cloud Operations (formerly Stackdriver)
Key management / HSM	KMS / CloudHSM	Key Vault / Managed HSM	Cloud KMS / Cloud HSM
Infrastructure as Code	CloudFormation / CDK	ARM Templates / Bicep / Terraform support	Deployment Manager / Config Connector / Terraform support
Cost & billing / tagging tools	Cost Explorer / Budgets / Tags	Cost Management + Billing / Tags	Billing Console / Cost Tools / Labels
Machine learning platforms	SageMaker	Azure ML	Vertex AI

```
$ qcc cloud describe-challenge --
attack-surface --service-count-all

> Calling agentic ai bot “provide service
  count by major cloud vendor”
```

Provider	Services
AWS	~ 430 services
Azure	~ 200–250 services
Google Cloud	~ 200–250 services

$ qcc cloud describe-challenge --attack-surface --compare-api-count

> Shoutout to https://github.com/iann0036/iam-dataset
> curl https://aws.permissions.cloud/, https://gcp.permissions.cloud/, and https://azure.permissions.cloud/

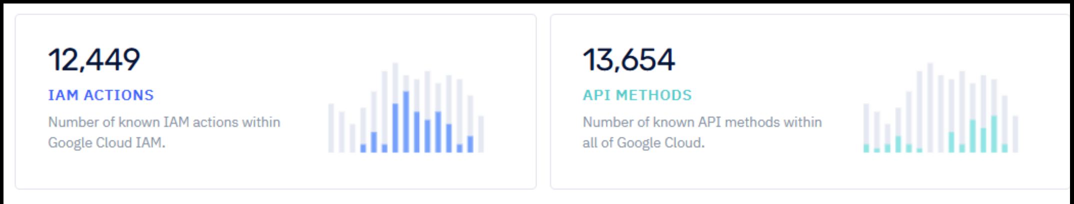


Global AWS Counts

● API METHODS ● IAM PERMISSIONS

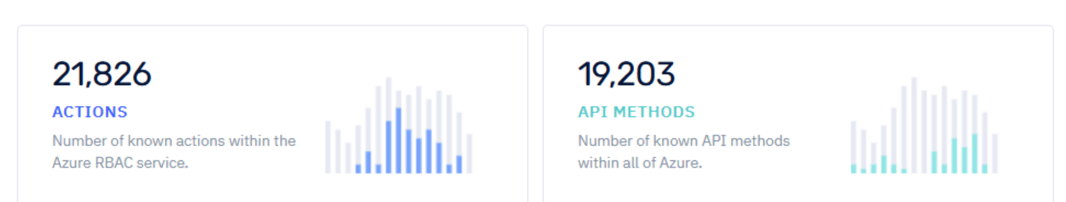
17,369
API METHODS

19,331
IAM PERMISSIONS



12,449
IAM ACTIONS
Number of known IAM actions within Google Cloud IAM.

13,654
API METHODS
Number of known API methods within all of Google Cloud.



21,826
ACTIONS
Number of known actions within the Azure RBAC service.

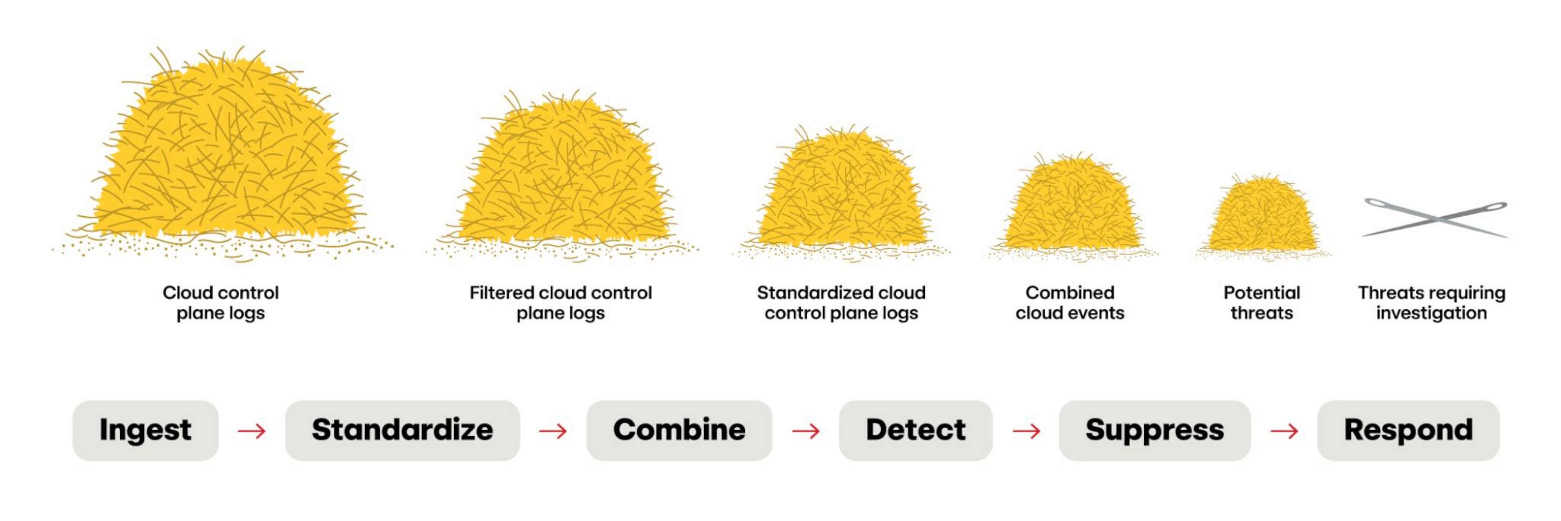
19,203
API METHODS
Number of known API methods within all of Azure.

```
$ qcc cloud list-mitigating-controls
>

   - Ignore and pray for the best
   - Capes & Unicorns
   - Cloud Detection and Response (CDR)
   - Don’t use cloud
```

```
$ qcc cdr get-definition

>

   Cloud Detection and Response: a
   cloud-native security approach
   designed to ingest, detect,
   enrich, and respond to cloud
   events
```

$ qcc cdr describe-process

```
> curl https://redcanary.com/cybersecurity-101/cloud-security/what-is-cloud-detection-and-response-cdr/
```



```
$ qcc cdr list-key-features

>
```

Real-Time Monitoring	Threat Intelligence Actionability
End-to-End Visibility	Graph-Based Analysis
Correlation at Scale	CSPM Data
Custom Response	Ecosystem Integration
API First	Auditability

```
$ qcc cdr get-value

>
```

1. Reduces alert fatigue
2. Accelerates response
3. Increases visibility and context

$ qcc cdr describe-address-challenges

>

Challenge	CDR Addresses It
Increased Attack Surface	Native cloud integration and continuously
Data Overload	Aggregates and normalizes cloud events
Dynamic Infrastructure	Auto-discovers new assets via logs and management apis
Fragmented Security	Unified visibility across multi-cloud environments
Evolving Threats	Integrations and api for automation.

```
$ qcc cdr compare --edr --ndr --xdr
```

```
> Calling agentic ai bot “how does cdr differ from edr, ndr, and xdr”
```

Technology	Scope
EDR (Endpoint Detection & Response)	Focused on hosts (servers, workstations)
NDR (Network Detection & Response)	Monitors network traffic
XDR (Extended Detection & Response)	Correlates across endpoints, network, email, etc.
CDR (Cloud Detection & Response)	Native understanding of cloud control plane, identity, and resource configurations

$ qcc cdr compare –soar –siem

```
> Calling agentic ai bot “how does cdr differ from soar and siem”
```

Technology	Scope
SOAR (Security Orchestration, Automation, and Response)	Automates playbooks and response
SIEM (Security Information and Event Management)	Centralized log collection & correlation
CDR (Cloud Detection & Response)	Native understanding of cloud control plane, identity, and resource configurations

```
$ qcc cdr get-initial-steps

>

  - Pick vendors
  - Identify key stakeholders
  - Create QFD
  - Proof of concepts (POC)
  - Pick
  - Profit
```

$ qcc cdr list-vendors

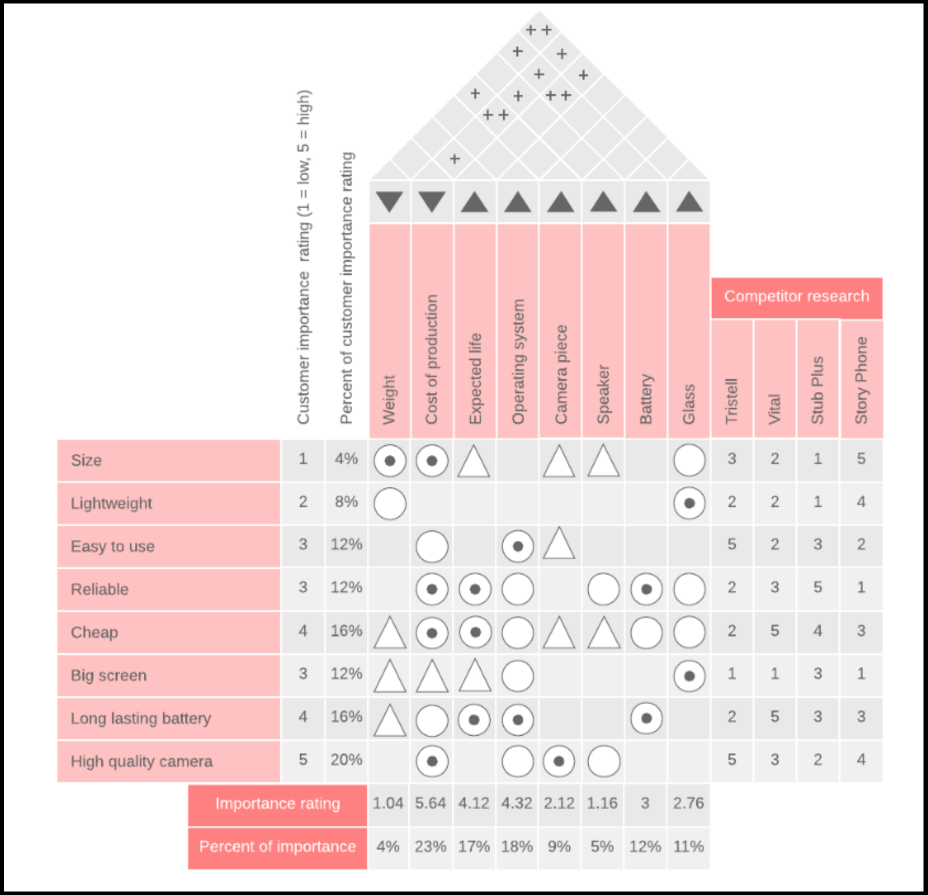
```
> curl letmegooglethat.com “cloud detection
  and response vendors”

   - Sysdig
   - Wiz
   - Crowdstrike
   - Palo Alto
   - Red Canary
```

```
$ qcc cdr list-stakeholders

> Calling agentic ai bot “potential internal
  company stakeholders for a CDR tool”

   - Threat Analyst Team
   - Cloud Platform Team
   - Governance/Risk Team
   - Audit Team
   - Identity Team
   - Architecture Team
```

```
$ qcc cdr describe-qfd

>

  QFD = Quality Function Deployment

  A structured framework for
  translating customer or
  stakeholder needs into
  measurable criteria.
```

$ qcc cdr get-qfd-example --traditional

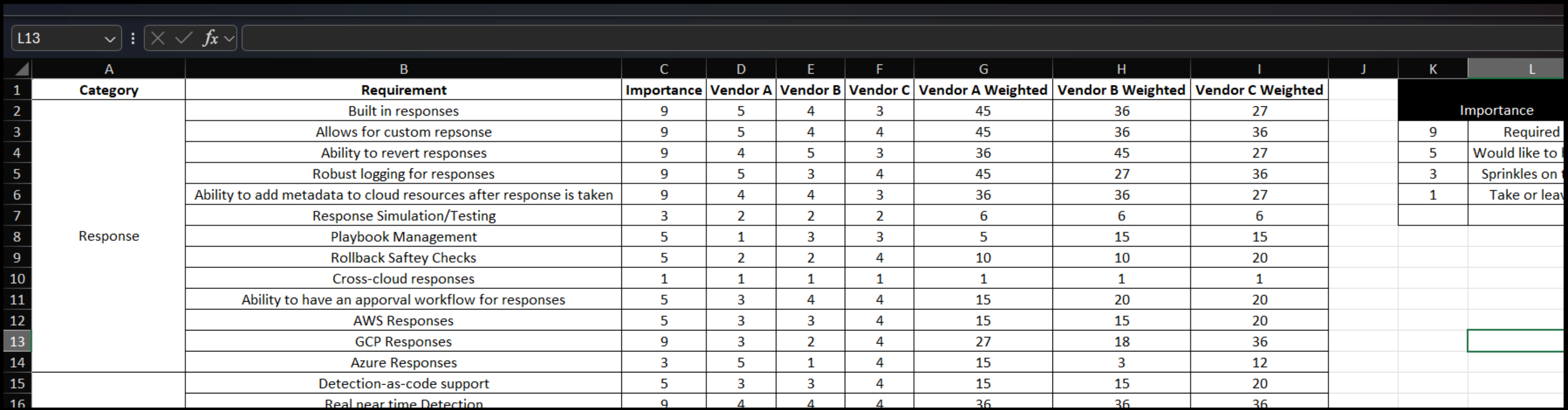
> curl https://www.lucidchart.com/blog/qfd-house-of-quality



											Competitor research			
	Customer importance rating (1 = low, 5 = high)	Percent of customer importance rating	Weight	Cost of production	Expected life	Operating system	Camera piece	Speaker	Battery	Glass	Tristell	Vital	Stub Plus	Story Phone
			▼	▼	▲	▲	▲	▲	▲	▲				
Size	1	4%	⊙	⊙	△		△	△		○	3	2	1	5
Lightweight	2	8%	○							⊙	2	2	1	4
Easy to use	3	12%		○		⊙	△				5	2	3	2
Reliable	3	12%		⊙	⊙	○		○	⊙	○	2	3	5	1
Cheap	4	16%	△	⊙	⊙	○	△	△	○	○	2	5	4	3
Big screen	3	12%	△	△	△	○				⊙	1	1	3	1
Long lasting battery	4	16%	△	○	⊙	⊙			⊙		2	5	3	3
High quality camera	5	20%		⊙		○	⊙	○			5	3	2	4
Importance rating			1.04	5.64	4.12	4.32	2.12	1.16	3	2.76				
Percent of importance			4%	23%	17%	18%	9%	5%	12%	11%				

$ qcc cdr get-qfd-example --modified

```
> curl https://github.com/infosec-shinobi/homelab_cdr/tree/main/assets/cdr_qfd.xlsx
```



L13

	A	B	C	D	E	F	G	H	I	J	K	L
1	**Category**	**Requirement**	**Importance**	**Vendor A**	**Vendor B**	**Vendor C**	**Vendor A Weighted**	**Vendor B Weighted**	**Vendor C Weighted**			
2	Response	Built in responses	9	5	4	3	45	36	27		Importance	
3		Allows for custom repsonse	9	5	4	4	45	36	36		9	Required
4		Ability to revert responses	9	4	5	3	36	45	27		5	Would like to
5		Robust logging for responses	9	5	3	4	45	27	36		3	Sprinkles on
6		Ability to add metadata to cloud resources after response is taken	9	4	4	3	36	36	27		1	Take or lea
7		Response Simulation/Testing	3	2	2	2	6	6	6			
8		Playbook Management	5	1	3	3	5	15	15			
9		Rollback Saftey Checks	5	2	2	4	10	10	20			
10		Cross-cloud responses	1	1	1	1	1	1	1			
11		Ability to have an apporval workflow for responses	5	3	4	4	15	20	20			
12		AWS Responses	5	3	3	4	15	15	20			
13		GCP Responses	9	3	2	4	27	18	36			
14		Azure Responses	3	5	1	4	15	3	12			
15		Detection-as-code support	5	3	3	4	15	15	20			
16		Real near time Detection	9	4	4	4	36	36	36			

$ qcc cdr describe-poc

```
> curl gitlab.local/cdr/poc_process.md

   1. Leverage QFD
   2. DIY
   3. Throw away environment
```

```
$ qcc cdr describe-implementation
> curl gitlab.local/cdr/implementation.md

  1. Partnerships
  2. Codify deployment
  3. Train and test
  4. Prioritize custom detections and responses
```

$ qcc cdr describe-best-practices

```
> curl gitlab.local/cdr/implementation.md

  - Codify all the things
  - Test/validation of detections/responses
  - Integrate tools for context enrichment and prioritization
  - Continuous monitoring
  - Partnerships
```

$ qcc cdr describe-next-steps

> Calling agentic ai bot “how do I mature my CDR tool deployment”

- Automated end to end testing
- Tool integrations
- MITRE Attack Framework Mapping
- Purple teaming
    - Simulated attacks
        - https://play.backdoorsandbreaches.com/play.backdoorsandbreaches.com-Engine-V1/App/?deck=Cloud+Security
        - https://stratus-red-team.cloud/
        - https://github.com/DataDog/grimoire
        - https://github.com/PaloAltoNetworks/cobra-tool
        - https://github.com/RhinoSecurityLabs/pacu
        - https://github.com/RhinoSecurityLabs/cloudgoat
        - https://github.com/nccgroup/sadcloud
- Red Team Ops

$ qcc cdr list-example-responses

```
> curl
  gitlab.local/cdr/custom_responses.md

   - Publicly Exposed S3 or Blob Storage
   Bucket Creation
   - Revert Suspicious Logging
   Modifications
   - Contain Identifies Assumed from
   Unknown Account
   - Network contain compute via firewall
   rules
    - Rotate/delete exposed keys
```

```
$ qcc cdr list-opensource

>

   - Threatmapper
   https://github.com/deepfence/ThreatMapper
   - ☹
```

```
$ qcc cdr describe-deployment -NimbusGuard
```

> curl https://github.com/infosec-shinobi/homelab_cdr/

*Not setup for production envs

```
$ qcc logout

> Executing order 66…

 All sessions terminated
```