Term Project: IT Security Policy for Tokenmetrics

ISA652
Nabeel Mahdi Abid Mehdi Sayed
G01213257

Table of Content

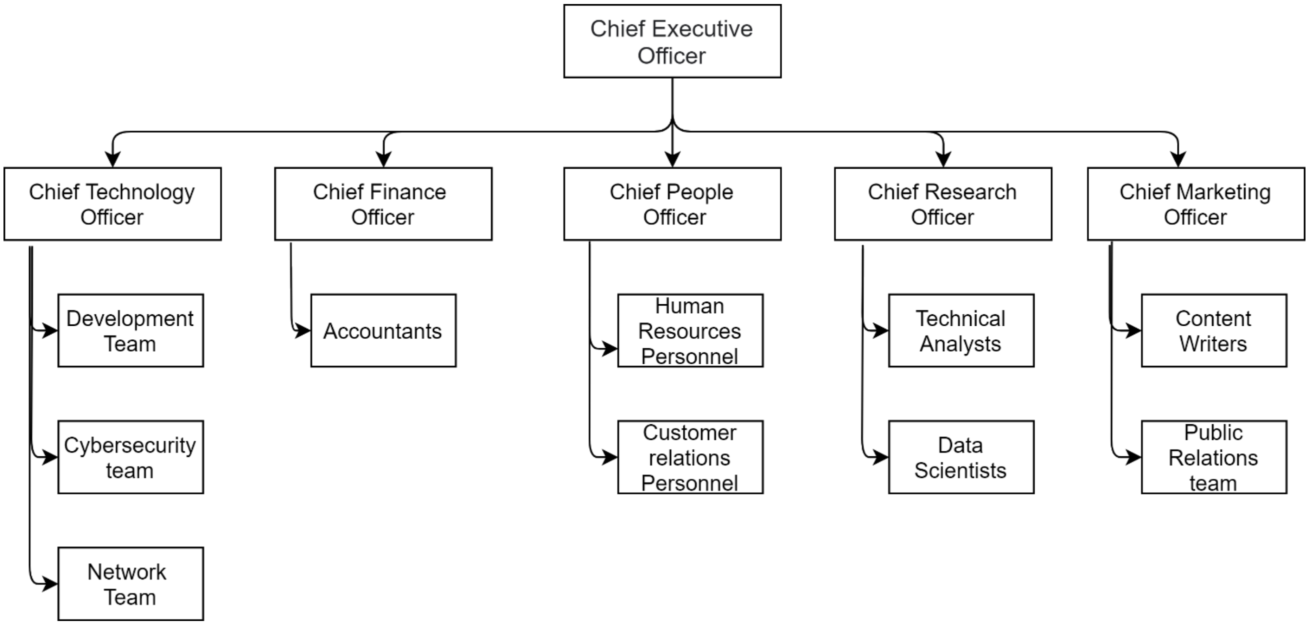
1. Introduction

My organization’s name is ‘Tokenmetrics’ and it is a crypto trading-based firm. Tokenmetrics is a digital currency wallet and platform where merchants and consumers can transact with new digital currencies like bitcoin, ethereum, and litecoin. The goal of the organization is making digital currency accessible and approachable for everyone. The organization provides a data-driven investment research platform for cryptocurrency, leveraging analytics and machine learning to help the investor.

The customer creates an account on the company’s website or mobile application. Once account is created, they put money in their digital currency wallet and then use that to buy and sell crypto currency. The organization also provides research report and analysis of market trends for its customers. In simpler term, we can say that ‘Tokenmetrics’ is a stockbroking platform for cryptocurrency.

2. Organizational Chart

the below chart depicts the current organizational structure. As you can see in the chart, there is CEO (Chief Executive Officer) on top. Below the CEO, there are other C-Suite officers. Each C-Suite User is head of his/her own department and deals with the employees and teams working under him/her.



3. Roles

In order to simplify the processes of understanding the organizational roles that employees of Token metrics hold, the organization has created a list of these roles, their symbols, and descriptions. The table below provides a thorough understanding of all roles within the organization.

Role Name	Symbol	Team	Description
User	USER		Generic user role given to anyone who interacts with our organization
Chief Executive Officer	CEO		CEO of the company
Chief Technology Officer	CTO	Technology	Head of technology department
Chief Finance Officer	CFO	Finance	Head of finance department
Chief People Officer	CPO	People	Head of people department. Takes care of human resources and customer relations
Chief Research Officer	CRO	Research	Head of research department
Chief Marketing Officer	CMO	Marketing	Head of marketing department
Development Team Member	DEV	Technology	Software developer
Cybersecurity Team Member	CSEC	Technology	Cybersecurity employee. Does cybersecurity related work
Network Team Member	NET	Technology	Performs network related tasks and administration of network resources
Accountant	ACC	Finance	Takes care of accounts and finance related things
Banker	BNKR		Performs transactions using company's bank account like paying or returning money
Human Resources Personnel	HR	People	Member of HR department. Takes care of hiring new individuals, keeping track of employee data and scheduling meetings
Customer relations Personnel	CRP	People	Member of customer relations department. Deals with the customer and works towards customer satisfaction.
Technical Analysts	TA	Research	Performs analysis on market and other research data and creates reports.
Data Scientists	DS	Research	Performs various algorithms on the data set and extracts information from it.
Content Writers	CW	Marketing	Writes content for customers
Public Relations Team Member	PR	Marketing	Deals with the public image of our organization.
Cloud Administrator	CLOUD		The administrator role for our cloud services account
Cloud Root	CROOT		This is the root account user for our cloud services account. This role is responsible for providing cloud administrator role to other users.
Customer	CUSTOMER		Customer of our company

4. Objects

The data systems of Tokenmetrics hold a number of objects which are protected by different levels of controls. These objects are thoroughly documented and require specific attention to the details of each regarding read and write access to them. The table below lists these objects.

Object Name	**Description**
Data files	These are files containing information or personal data that is being used by our organization.
Logs	These are the log files stored in various locations in our system
Source code	These are the files containing source code of our organization's IT infrastructure.
Encryption keys	These are the encryption keys used by our organization to perform cryptographic operation
Firewall	The firewall being used
Database	The database being used by our organization
Intrusion detection and prevention system	The configuration and admin page or our IPS and IDS.
Finance & legal doc	This object includes spreadsheets containing company transactions, data on customer accounts, tax returns, legal documents and other financial documents
Bank Account	Company's Bank Account
Cloud resources	Cloud resources used by our organization
Meeting	These are the meetings schedule by human resources personnel
Employee Data	This a database which contains information on each employee. This information includes leaves, salary, cloud resources used, roles assigned and other information.

5. Policy Description

The IT security policy structure at Tokenmetrics has been designed in such a manner that the utmost security is obtained but with simple understandability and ease for the users. Given below are few examples that justify the use of various policy structures.

5.1. Discretionary Access Control

5.1.1.

Whenever users are dissatisfied or they want some information or want to make changes to their account, they will contact the Customer Relations Personnel. Customer Relations Personnel should be able to retrieve information of user accounts when dealing with customers. Customers will identify themselves and customer relations personnel will make changes to customer accounts according to customer's need. Here, customers do not directly interact with their data files, instead customer relations personnel act as a middleman. We cannot use a MAC approach here because each customer user owns his own account details, and a hierarchy-based approach would complicate the process of ownership of account files.

Policy:

In this scenario, a customer relations personnel will allow a user to do operations.
Let Subjects be S = {s1, s2, s3….sn | i∈N},
And Objects be O= {o1, o2, o3…on | i∈N }
Here objects are account details of a user which are stored as objects in different database tables.

Let Permission be, P(s,o) = {w, r, o, Ø}, where p(s,o) tells permission of subject s on object o
r = read, and w = write, o = own

Let Roles be R(s) = {r[customer], r[crp], r[hr] …, rn},
here, R(s1) is the role assigned to subject s1 and r[CRP]= customer relations personnel role

and s1(o1, {r,w}), means that s1 is doing read and write operation on o1

and 's1 → operation', describes that subject s1 is the one responsible for doing the operation.

Solution:

∀ s ∈ S, R(s1)=r[CRP], R(s2)=r[CUSTOMER]

```
s1      →      ALLOW s1(o1, {r,w})
               if P(s2,o1) == {o}
               else DENY
```

5.1.2.

Our company uses a cloud infrastructure and utilizes plenty of Amazon's cloud services. The teams working under CTO will require access to various cloud services depending on their needs. Since the access required for each team is very diverse, using MAC based approach will not be feasible. MAC requires a privilege boundary and in this scenario the rights for services are very complex.

Solution:

Objects	Development team (DEV)	Cybersecurity team (CSEC)	Network team (NET)
logs		Create, Read	Create, Read
Source code	Create, Modify, Read	Read, Modify	
Encryption keys	Create, Read	Read	Read
Firewall		Modify, Read	Create, Modify, Read
Database	Create, Modify, Read	Read	
Intrusion detection and prevention system		Create, Modify, Read	

5.2. Secrecy-Based Mandatory Access Control

5.2.1.

We have a cybersecurity team in place which requires access to a lot of files and resources to test the integrity of IT infrastructure. But as we know in a large organization there are plenty of files that store confidential information like company secrets, sensitive passwords and important information stored in various files.
In order to make sure the security team works efficiently without leaking important information we require a MAC based system where each level has its own set of accesses to files. In such a scenario, using DAC will not be convenient because lower clearance level-based users are not meant to access the files for higher levels.

<u>Policy:</u>

R = role (CSEC), CSEC= Cybersecurity team member
S = subject
O = object (data files)
P= permissions, where r = read, w = write
C = classification, where C = confidential, S = Secret, TS = Top Secret
And C(s) is classification of s
And $TS>S>C$

<u>Solution:</u>

$\forall$ s $\in$ S, R(s1) = r[CSEC], o1=data files

```
s1      →      ALLOW s1(o1, r)
               if $c(s1) > c(o1)$
               else DENY

s1      →      ALLOW s1(o1, w)
               if $c(s1) < c(o1)$
               else DENY

s1      →      ALLOW s1(o1, {r,w})
               if $c(s1) = c(o1)$
               else DENY
```

5.2.2.

The accountants working for our organization will have to deal with a lot of spreadsheets and legal documents including tax returns and customer account data. Our organization is a trading firm and will be performing a lot of transactions on behalf of its customers, as well as there will be transactions with banks and other service-based companies. Among these transactions, few transactions can be of high importance and leakage of such information can cause problems like insider trading and data leakage. To keep company transactions secret, it is a good approach to employ MAC here.

The transaction spreadsheets and databases will be given clearance levels and the accesses to these files will be done using MAC. DAC will not be a good approach here because lower clearance-level users are officially not even meant to know the existence those transaction files and databases.

Policy:

R = role (ACC), ACC= Accountant

S = subject

O = object (Accounts & legal doc)

P= permissions, where r = read, w = write

C = classification, where C = confidential, S = Secret, TS = Top Secret

And C(s) is classification of s

And TS>S>C

Solution:

```
∀ s ∈ S, R(s1) = r[ACC], o1= Accounts & legal doc

s1      →      ALLOW s1(o1, r)
               if c(s1) > c(o1)
               else DENY

s1      →      ALLOW s1(o1, w)
               if c(s1) < c(o1)
               else DENY

s1      →      ALLOW s1(o1, {r,w})
               if c(s1) = c(o1)
               else DENY
```

5.3. Integrity-Based Mandatory Access Control

5.3.1.

In our organization, there is a flow of information which starts from data scientists then goes to technical analysts and then ends in the hand of content writers. Basically, data scientists apply their algorithms on the market data and generate a refined set of information, now this information is used by the technical analysts to come up with patterns, estimations and predictions in form of a report. These reports made by technical analysts are used by content writers to write blogs for their customers in order to explain the findings of the research team in a better way.

In this scenario we have three teams in hand, data scientists, technical analysts and content writers. And all three are dependent on each other. Content writers should not be allowed to change the reports and files generated by data scientists and technical analysts, but they should be able to read it. At the same time technical analysts should not be able to change the files and reports generated by data scientists. But data scientist should be able to correct reports generated by the content writers and technical analysts because they understand the data and reports better than the levels below. Using DAC would be less appropriate here as individual permission might deny few users at data scientist level to review the work done by technical analysts.

Policy:

R = role (DS, TA, CW) , DS=data scientists, TA=technical analysts, CW=content writers
S = subject (user)
O = object (data files)
P= permissions, where r = read, w = write
C = classification, where D = data scientist, T = Technical analyst, C = Content Writer
And C(s) is classification of s
And D>T>C

Solution:

∀ s ∈ S, R(s1)=r[DS, TA, CW], o1=data files

s1 → ALLOW s1(o1, w)
if c(s1) > c(o1)
else DENY

s1 → ALLOW s1(o1, r)
if c(s1) < c(o1)
else DENY

s1 → ALLOW s1(o1, {r,w})
if c(s1) = c(o1)
else DENY

5.3.2.

Another good example of MAC would be developers. All developers will be given different level of access to the source files depending on the importance. The important code segment will be given to the programmers at the highest level and things like User interface and less important segments of the code can be given to lower level developers. Using DAC would be less appropriate here as individual permission might deny developers at higher level to review any changes made at a lower level.

<u>Policy:</u>

R = role (DEV), DEV=Developer
S = subject (user)
O = object (source code)
P= permissions, where r = read, w = write

C = classification, where TS = Top Secret, S = Secret, U = Unclassified
And C(s) is classification of subject s
And TS>S>U

<u>Solution:</u>

$\forall$ s $\in$ S, R(s1)=r[DEV], o1=source code

```
s1      →      ALLOW s1(o1, w)
               if c(s1) > c(o1)
               else DENY

s1      →      ALLOW s1(o1, r)
               if c(s1) < c(o1)
               else DENY

s1      →      ALLOW s1(o1, {r,w})
               if c(s1) = c(o1)
               else DENY
```

5.4. Role-Based Access Control

5.4.1.

Accountants will be using a ‘banker’ role to move money in and out of accounts. It is possible that when customer relations individual interacts with the customers, the customer might ask for information regarding a transaction (credit card used, time, etc) or request refund for an invalid subscription. In such a scenario, the customer relations personnel would need access to this ‘banker’ role to return the money. Banker role allows our users to move money in and out of the accounts.

Policy:

R = role (BNKR, ACC, CRP), BNKR=banker, ACC=accountant, CRP=customer relation personnel
S = subject (user)
O = object (back account)
P= permissions, where r = read, m = modify

s1(o1, {r,m}), means that s1 is doing read and modify operation on o1

Solution

∀ s ∈ S, R(s1) = r[ACC, CRP], o1=bank account

s1 → R(s1) = r[BNKR]

s1 → s1(o1, {r,m})

5.4.2.

As we know, our IT infrastructure is based on cloud and different teams in our company will have different accesses to the cloud account. Developers might require to get new cloud services or increase the current cloud resources, the network team might require to increase network size and in a similar manner, other teams might require to get new resources or make changes to their current cloud resources. For this purpose, we need a cloud admin role which allows us to add new cloud services, make changes to the current services and increase the cloud resources.

Policy:

R = role (CLOUD), CLOUD=Cloud admin
S = subject (user)
O = object (cloud resources)
P = permission, where m = modify resources in hand, a = add new service, i = increase resources

s1(o1, {i}), means that s1 is doing increase resource operation on cloud service o1

Solution:

∀ s ∈ S, R(s1) = r[CLOUD], o1=cloud resources

s1 → ALLOW s1(o1, {m,a,i})

5.5. Attribute Based Access Control

5.5.1.

A good example of Attribute based access policy would be that of cybersecurity team. Where accesses to certain files are decided using the access level assigned to the file. Here the access level of a file is an attribute which we will be using to grant or deny access. For this situation RBAC alone does not provide enough attribute identification and specification for proper permissions to be assigned. These attributes transcend the ability of a traditional ACL to identify.

<u>Policy:</u>

R = role (CSEC), CSEC=Cybersecurity team member
S = subject (user)
O = object (data files)
P= permissions, where r = read, w = write

C = classification, where C = confidential, S = Secret, TS = Top Secret
And C(s) is classification of s
And TS>S>C

s1(o1, {r}), means that s1 is doing read on o1

<u>Solution:</u>

$\forall$ s $\in$ S, R(s1)=r[CSEC], o1=data files

```
s1    →    ALLOW s1(o1, r)
           if c(s1) > c(o1)
           else DENY
s1    →    ALLOW s1(o1, w)
           if c(s1) < c(o1)
           else DENY
s1    →    ALLOW s1(o1, {r,w})
           if c(s1) = c(o1)
           else DENY
```

5.5.2.

The human resources personnel will have to schedule meetings for executive team, hiring new employees or project planning. The needs of such meetings might be different, but these meetings should be able to be attended by only a specific set of people in a specific period of time. For an example, there can be a meeting for only the C-suite scheduled between 9am to 10am on Monday morning and no one else should be allowed to sit for this meeting since the topics of this meeting are company secrets. In such a scenario, using an RBAC for the meeting will not be a feasible solution since there is the meeting time as well which has to be considered apart from who all are attending. ACL is also not feasible here because of the time attribute since ACL can consider only accesses to the meeting not the time duration.

Policy:

Let Subjects be S = {s1, s2, s3….sn | i∈N},
There are our users

And Objects be O= {o1, o2, o3…on | i∈N }
Here objects are the meetings.

And time periods be T= {t1, t2, t3…tn | i∈N }

Let G be a group of subjects like G=[s1, s2], here s1 and s2 are part of G
And s1 ∈ g1 denotes that s1 us part of group g1

and o1(t1, g1) mean that meeting o1 can be attended at t1 by group g1

Let Roles be R(s) = {r[CUSTOMER], r[CRP], r[FR], …, rn | i∈N },
here, R(s1) is the role assigned to subject s1

and s1(o1) mean that subject s1 can attend meeting o1

and 's1 → operation', describes that subject s1 is the one responsible for doing the operation.

Solution:

∀ s ∈ S, R(s1) =r[HR], R(s2)=r[USER], t1= meeting time, t2=current time, o1=Meeting

```
s1      →      ALLOW s1(o1)
               if t2==t1 AND s1 ∈ g1 AND o1(t1, g1)
```

5.6. Separation of Duties

For the purpose of audit, human resources team keeps information on each employee like leaves taken, salary, company transactions, meetings, cloud logins, amount of cloud resources requested and etc. In order to make sure that HR personnel do not change their own data the entire organization is divided into 4 teams (Alpha, Beta, Gamma, Delta). The HR personnel are also assigned a team and they cannot write, modify or read the data belonging to their own team to make sure that they do not change their own data.

Policy:

Let teams be T= {Alpha, Beta, Gamma, Delta}
Here, T(s1) is the team of subject s1

Let Subjects be S = {s1, s2, s3….sn | i∈N},
Here subjects are users

And Objects be O= {o1, o2, o3…on | i∈N }
Here objects are employee data

T(o1) ∈ T(s1) means that data stored in object o1 belongs to same team as s1
And T(o1) ∉ T(s1) means that o1 and s1 belong to different teams.

Let Roles be R(s) = {r[CUSTOMER], r[CRP], r[HR], …, rn | i∈N },
here, R(s1) is the role assigned to subject s1

s1(o1, {r,w}), means that s1 is doing read and modify operation on o1

Solution:

∀ s ∈ S, R(s1) = r[HR], o1=Employee data

```
s1    →    ALLOW s1(o1, {r,w})
           if T(o1) ∉ T(s1)
           else DENY
```

5.7. Hierarchical Relationship

We have a huge organization with a CEO on top. The CEO will have to take decisions for the sake of our organization. The CEO will not be going to individual employee and giving them tasks, instead he will be assigning tasks to the other C-Suite employees. The C-Suite will then assign subtasks to different teams in the same department. The task assigned are more of a goal which the team must achieve.
There are 5 departments in our organization namely technology department, finance department, people's department, research department and marketing department. On top of each such department is a C-Suite employee excluding CEO.

<u>Policy:</u>

Let Departments be D= {technology, finance, people, research, marketing}
And D(s) is the department for the employee. (D(s)= Ø for CEO)

Let t be task/goal assigned such that T={t1,t2,t3…tn}

Let roles be R={CEO, CS, GE} where CEO= Chief executive officer, CS= Remaining C-Suite (CTO, CFO, CMO, CRO, CPO), GE=General Employee (Anyone not in C-Suite)

And R(s) is the role assigned to s

Let s be the subjects such S= {s1,s2,s3….sn}

And s1(s2, t) implies that subject s1 can assign task t to subject s2

<u>Solution:</u>

∀ s ∈ S, R(s1) = r[USER], t1=task

```
s1    →    ALLOW s1(s2, t)
           if (R(s1) == CEO AND R(s2)== CS) OR (R(s1)== CEO AND R(s2)== GE)
           OR (R(s1) == CS AND R(s2)== GE AND D(s1)==D(s2))
           else DENY
```

5.8. Positive and Negative Permissions

Our entire organization follows a closed policy or positive authorization. Due to the use of a closed system, negative permissions are not used, as they would prove extremely time-intensive to apply to every user in an open system. For our organization 'Most-specific-takes-precedence' type of conflict resolution policy should be used since we use a default deny for all roles. The roles that have been specifically assigned privileges for specific objects should be allowed to keep those privileges.

A good example for positive policy would be 'cloud admin' role. As we know, a normal user has no right to make changes to cloud resources at hand. Only after obtaining a 'cloud admin' role, the user can use this role's privileges and make changes to cloud resource.

<u>Policy:</u>

R = role (CLOUD, USER), CLOUD=Cloud administrator, USER=Normal user
S = subject (user)
O = object (cloud resources)
P = permission, where m = modify resources in hand, a = add new service, i = increase resources

s1(o1, {i}), means that s1 is doing increase resource operation on cloud service o1

<u>Solution:</u>

For a normal user:

∀ s ∈ S, R(s1) = r[USER]
s1 → ALLOW s1(Ø, Ø)

For cloud admin user:

∀ s ∈ S, R(s1) = r[CLOUD] AND r[USER], o1=cloud resources
s1 → ALLOW s1(o1, {m,a,i})

5.9. Temporal Authorization

Providing credentials for root account of the AWS cloud account to any one is not a good decision. It is possible that our organization might hire a consultant, an advisor, or an auditor who might need the credentials to this account. Providing direct credentials for this account as it is, is not a smart decision. In such scenarios, the root account administrator can do the provision of temporary access credentials which can be used on temporary basis by people who need them. AWS has an IAM service which provides temporary secret key that can be used to gain a one-time access to a role for a limited amount of time.

Policy:

Let Subjects be S = {s1, s2, s3….sn | i∈N},
Here subject s is our user

And Objects be O= {o1, o2, o3…on | i∈N }
Here objects are AWS resources.

And s(o) means that subject s can use resource o

And time frames be T= {t1, t2, t3…tn | i∈N }

Let TT be Temporary Token such that, TT(s) = True, means that subject s has a temporary access token.

Let Roles be R(s) = {r[CUSTOMER], r[CRP], r[HR], r[CLOUD], r[CROOT] …, rn | i∈N },
CROOT= Cloud root account administratot
here, R(s1) is the role assigned to subject s1

Solution

∀ s ∈ S, R(s1) = r[CROOT], t1= temporary time frame allotted to the role, t2=current time, o1=cloud resources

```
s1    →    ALLOW s1(o1)
           if TT(s1) == True AND t2 == t1
           else DENY
```

5.10. Administrative policy

The administrative policy used in our organization is hierarchical and centralized. In order to understand why such a combination of a policy is being used, let us go through the use case of each of these policies in our organization.

In section 7 we see a structure where tasks are given by the CEO to the remaining C-Suite, and then this subsection of C-suite assigns task to the employees working below them. If you observe, you will find the entire structure of giving tasks follows a hierarchical pattern. Due to this we can state that our organization is following a hierarchical policy as far as section 7 is concerned.

For all other cases (Section 1,2,3,4,5,6) we are assigning roles, teams and clearance levels to the users in our organization. This process of assignment of roles is centralized and done by the C-Suite Users. Since there is only one body that is assigning such roles, teams and clearance level, using a centralized policy seems fit for this scenario.

As we saw above, in different use cases in our organization, we are either using a centralized or hierarchical policy. Therefore, the administrative policy used in our organization is hierarchical and centralized

6. Tokenmetrics Security Audit Program

COBIT 4.2 Domain	**Control Category**	**Policy/Procedure**	**Test Considerations**	**COBIT 4.2 Sub Domains**
DS11 Manage Data PO4 Define IT Processes, organization and relationship PO2 Define Information Architecture DS8 Manage Service Desk and Incidents DS5 Ensure Systems Security	Discretionary Access Control (DAC)	• $s \in S$, $R(s1)=r[crp]$, $R(s2)=r[customer]$ $1 \rightarrow$ ALLOW $s1(o1, \{r,w\})$ if $P(s2,o1) == \{o\}$ else DENY • Section 5.1.2 talks about users who can create, modify or use encryption keys and it is in compliance with DS5.8 • Section 5.1.1 talks about how a customer has to approach a customer relations personnel to get information or make changes to their account. User account management is done through the customer relations personnel and this is in compliance with DS 8.1 and DS5.4	• Validate that the customer relations personnel have the ability change to customer's account files upon request. • Validate that all user roles have been allocated correct permissions to access the cloud resource. • Enquire about the existence of standards of service and communication of the standards with customers. • Make sure that a policy has been defined and implemented to protect unauthorized access and tampering of cloud resources. • Validate that only owner of the user file can get access to the file through customer relations personnel. • Validate that the encryption keys can only be used by the users with access to it and the user with access can not grant anyone else access to the key. • Validate that customer who has confirmed his identity will get all the privileges of his role through customer relations personnel. • Make sure that customer relations personnel register the customer's query in a database or an excel file. • Enquire about procedures to grant cloud resource accesses to new users assigned to a role.	DS11.6 Security Requirements for Data Management PO4.9 Data and System Ownership PO2.4 Integrity Management DS8.1 Service Desk DS5.8 Cryptographic Key Management DS5.4 User Account Management

COBIT 4.2 Domain	**Control Category**	**Policy/Procedure**	**Test Considerations**	**COBIT 4.2 Sub Domains**
PO2 Define Information Architecture PO4 Define IT Processes, organization and relationships DS11 Manage Data AC3 Accuracy, Completeness and Authenticity Checks	Secrecy based Mandatory Access Control	• $\forall s \in S$, $R(s1)=r$[cybersecurity user] $s1 \rightarrow$ ALLOW $s1(o1, r)$ if $c(s1) > c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, w)$ if $c(s1) < c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, \{r,w\})$ if $c(s1) = c(o1)$ else DENY • $\forall s \in S$, $R(s1) = r$[Accountant] $s1 \rightarrow$ ALLOW $s1(o1, r)$ if $c(s1) > c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, w)$ if $c(s1) < c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, \{r,w\})$ if $c(s1) = c(o1)$ else DENY	• Validate that if a user has need to know and is classified at a higher level than a file or document, then the user will only have read access. • Validate that if a user is classified at a level less than the file or document, that the user can obtain write access only. • Validate that if the secrecy classification level of a subject and an object in the transactional and legal document is equal then the subject can read and write to the object. • Establish a classification scheme that is applied throughout the organization. • Establish correct roles and clearance levels associated with the role. • Define and implement policies and procedures to apply security requirements to the data.	PO2.3 Data Classification Scheme PO4.6 Establishment of Roles and Responsibilities AC3 Accuracy, Completeness and Authenticity Check DS11.6 Security Requirements for Data Management
PO2 Define Information Architecture AC4 Processing Integrity and Validity PO4 Define the IT Processes, Organization and Relationships	Integrity Based Mandatory Access Control	• $\forall s \in S$, $R(s1)=r$[data scientists, technical analysts, content writers] $s1 \rightarrow$ ALLOW $s1(o1, w)$ if $c(s1) > c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, r)$ if $c(s1) < c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, \{r,w\})$ if $c(s1) = c(o1)$ else DENY • $\forall s \in S$, $R(s1)=r$[developer] $s1 \rightarrow$ ALLOW $s1(o1, w)$ if $c(s1) > c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, r)$ if $c(s1) < c(o1)$ else DENY $s1 \rightarrow$ ALLOW $s1(o1, \{r,w\})$ if $c(s1) = c(o1)$ else DENY	• Validate that if a subject's integrity classification is equal to that of an object, and then the subject can both read and write to that object. • Validate that if a subject's integrity classification is greater than that of an object, it can only write to that object. • Validate that if a subject's integrity classification is less than that of an object, then the subject can only read to that object • Confirm through interviews that supervisory practices have been established for data scientists and	PO2.4 Integrity Management PO2.3 Data Classification Scheme AC4 Processing Integrity and Validity PO4.10 Supervision

COBIT 4.2 Domain	Control Category	Policy/Procedure	Test Considerations	COBIT 4.2 Sub Domains
		• In section 5.3.1, the user roles at upper level are responsible for overlooking and correcting the work done at lower levels and is in compliance with PO4.10	technical analysts. • Review records to assess the frequency and extent of supervisory review	
PO4 Define the IT Processes, Organisation and Relationships DS3 Manage Performance and Capacity AI3 Acquire and Maintain Technology Infrastructure DS11 Manage Data	Role Based Access Control	• $\forall s \in S$, R(s1)=r[accountant, customer relation personnel], o1=bank account s1 → R(s1) = r[banker] s1 → s1(o1, {r,m}) • $\forall s \in S$, R(s1) = r[cloud admin], o1=cloud resources s1 → ALLOW s1(o1, {m,a,i}) • Section 5.4.2 is in compliance with AI13.1 and DS3.4 because it allows our users to increase the resources as per the need to maintain availability and allows acquisition of new resources when needed.	• Enquire of key staff members about the process to obtain, review and implement vendor requirements, and confirm that the current capacity and performance capabilities have incorporated the vendor requirements • Enquire of management for known performance and capacity gaps. • Enquire of key staff members about the process to correct performance and capacity issues • Confirm with staff members that they are aware of using the cloud admin role for the acquisition and upgrade of the technology infrastructure. • Ensure that users with ‘banker’ role can access the company accounts and make necessary changes. • Make sure that only users with right roles can access the privileges assigned to that role. • Make sure that the roles are given proper privileges	PO4.6 Establishment of Roles and Responsibilities DS3.4 IT Resources Availability AI3.1 Technological Infrastructure Acquisition Plan DS11.6 Security Requirements for Data Management

COBIT 4.2 Domain	**Control Category**	**Policy/Procedure**	**Test Considerations**	**COBIT 4.2 Sub Domains**
PO2 Define the Information Architecture PO6 Communicate Management Aims and Direction AC3 Accuracy, Completeness and Authenticity Checks AI4 Enable Operation and Use	Attribute Based Access Control	• $\forall$ s $\in$ S, R(s1)=r[cybersecurity user] s1 $\rightarrow$ ALLOW s1(o1, r) if c(s1) > c(o1) else DENY s1 $\rightarrow$ ALLOW s1(o1, w) if c(s1) < c(o1) else DENY s1 $\rightarrow$ ALLOW s1(o1, {r,w}) if c(s1) = c(o1) else DENY • $\forall$ s $\in$ S, R(s1)=r[Human Resources Personnel], R(s2)=r[user], t1= meeting time, t2=current time s1 $\rightarrow$ ALLOW s1(o1) if t2==t1 AND s1 $\in$ g1 AND o1(t1, g1) • Section 5.5.2 is in compliance with PO6.5, AI4.2 and AI4.4 since through meetings necessary individuals are informed about various things. Communication and knowledge transfer is taking place through meetings.	• Establish correct clearance levels associated with the role. • Establish correct attributes are associated with the objects and only authorized individuals can access them • Make sure that only authorized individuals can attend a meeting. • Make sure that the meeting can only be attended at a specific time period • Make sure that key business objectives are communicated in the meeting • Enquire whether and confirm that processes and procedures are established for the segregation of duties for entry, modification and approval of transaction data as well as for validation rules • Enquire whether and confirm that validation criteria and parameters on input data are periodically reviewed. • Interview key staff members about the user group's awareness and knowledge of the meeting process	PO2.3 Data Classification Scheme PO6.5 Communication of IT Objectives and Direction AC3 Accuracy, Completeness and Authenticity Check AI4.2 Knowledge Transfer to Business Management AI4.4 Knowledge Transfer to Operations and Support Staff
PO2 Define the Information Architecture PO4 Define the IT Processes, Organisation and Relationships DS11 Manage Data	Separation of Duties	• $\forall$ s $\in$ S, R(s1) = r[Human Resources Personnel] s1 $\rightarrow$ ALLOW s1(o1, {r,w}) if T(o1) $\notin$ T(s1) else DENY For further explanation refer to section 5.6	• Validate that when human resources personnel are assigned to a specific team, then the users cannot access their own team's database. • Make sure that each member in human resources is assigned to team. • Validate no one apart from human resources personnel can access this	PO2.3 Data Classification Scheme PO2.4 Integrity Management PO4.11

COBIT 4.2 Domain	**Control Category**	**Policy/Procedure**	**Test Considerations**	**COBIT 4.2 Sub Domains**
			database. • Validate that each data file for user is classified to a team. • Obtain all the user related data and store it in a centralized location/database. • For each data element, confirm that requirements for confidentiality, integrity and availability have been defined and that these requirements have been validated with the data owners. • Ensure that requirements have been established for physical and logical access to data	Segregation of Duties DS11.1 Business Requirements for Data Management DS11.6 Security Requirements for Data Management
PO4 – Define the IT Processes, Organization and Relationships AI4 Enable Operation and Use	Hierarchical relationships	For a full explanation of this policy/procedure, please see section 5.7.	•Validate that periodic reviews are conducted to measure the impact of organizational change as that affects the overall organization and structure of IT function. • Validate that roles and responsibilities have been established for IT personnel and end users to communicate and that delineate between IT personnel and end user authority, responsibilities and accountability for meeting the organization's needs. • Role descriptions for staff members across the organization specifically identify responsibilities regarding information systems, internal control and security. • Implement adequate supervisory practices in the IT function to ensure that roles and	PO4.5 IT Organizational Structure PO4.6 Establishment of Roles and Responsibilities PO4.10 Supervision PO4.15 Relationships AI4.2 Knowledge Transfer to

COBIT 4.2 Domain	Control Category	Policy/Procedure	Test Considerations	COBIT 4.2 Sub Domains
			responsibilities are properly exercised. • Establish a mechanism to communicate the information to each level. • Make sure that each user is updated on the goal assigned to him	Business Management AI4.4 Knowledge Transfer to Operations and Support Staff
PO4 Define the IT Processes, Organization and Relationships AI3 Acquire and Maintain Technology Infrastructure DS3 Manage Performance and Capacity	Positive and Negative Permissions	For a full explanation of this policy/procedure, please see section 5.8.	• Inspect the cloud configuration to confirm that key aspects have been addressed, including the modification of default passwords, initial application parameter settings relative to security and any other vendor defaults • Enquire whether and confirm that temporary access granted to cloud admin account is monitored and that passwords are changed immediately • Confirm with key staff members that access to sensitive infrastructure components is logged and regularly reviewed • Enquire of key staff members about the process to obtain, review and implement vendor requirements, and confirm that the current capacity and performance capabilities have incorporated the vendor requirements • Enquire of management for known performance and capacity gaps. • Enquire of key staff members about the process to correct performance and capacity issues	PO4.6 Establishment of Roles and Responsibilities AI3.1 Technological Infrastructure Acquisition Plan AI3.2 Infrastructure Resource Protection and Availability DS3.4 IT Resources Availability

COBIT 4.2 Domain	Control Category	Policy/Procedure	Test Considerations	COBIT 4.2 Sub Domains
			• Confirm with staff members that they are aware of using the cloud admin role for the acquisition and upgrade of the technology infrastructure.	
PO4 Define the IT Processes, Organization and Relationships AI3 Acquire and Maintain Technology Infrastructure DS13 Manage Operations	Temporal Authorization	• $\forall$ s $\in$ S, R(s1) = r[CROOT], t1= temporary time frame allotted to the role, t2=current time, o1=cloud resources s1 → ALLOW s1(o1) if TT(s1) == True AND t2 == t1 else DENY • Section 5.9 is in compliance with PO4.14 and 13.1 since the roles are being used by external users to do tasks in our organization and they need to comply with our organization's policy for the temporary credential use of root account. • Section 5.9 is compliant with DS 13.4 because a temporary token with a restraint is being provided to the user. • Section 5.9 is also compliant with DS13.2 since access to root account is scheduled according to our organization's needs	• Ensure that the root account admin has the right privileges to provide the temporary token. • Ensure the security of temporary token • Ensure the temporary root account role has all the necessary privileges • Ensure that the external user is complying to our organization's policies • Ensure that procedures exist to remove and dispose temporary token. • Enquire whether and confirm that temporary access granted to cloud root account is monitored and logged • Ensure that the temporary access has a scheduled start and end time • Ensure that period assigned for temporary access does not coincide with any critical task for the root account user • Keep a copy of standard operational procedure for temporary access to root account and make sure the external user follows it	PO4.6 Establishment of Roles and Responsibilities PO4.14 Contracted Staff Policies and Procedures AI3.2 Infrastructure Resource Protection and Availability DS13.4 Sensitive Documents and Output Devices DS13.1 Operations Procedures and Instructions DS13.2 Job Scheduling

COBIT 4.2 Domain	**Control Category**	**Policy/Procedure**	**Test Considerations**	**COBIT 4.2 Sub Domains**
PO1 Define a Strategic IT Plan PO4 Define the IT Processes, Organization and Relationships DS13 Manage Operations ME4 Provide IT Governance	Administrative policy	For a full explanation of this policy/procedure, please see section 5.10.	• Standard IT operational procedures covering the definition of roles and responsibilities, including those of external service providers exist and are maintained. • Procedures and responsibilities for formal handover of duties are documented • Appropriate management structures such as an IT steering committee, technology council, IT architecture review board and IT audit committee exist • Enquire whether and confirm that a process for identifying stakeholders has been defined and that a communication channel have been established for each to make sure hierarchical policy is implemented correctly • Review that an IT strategy committee is in place to assign roles, teams and clearance levels to various users in our organization. • All IT governance issues and remedial actions into a consolidated management context for reporting is implemented. Report to the board the status of IT governance issues and activities and identify their impact on strategic initiatives and enterprise outcomes. • IT investment portfolio management disciplines exist, which include periodic review of portfolios to verify their continued relevance to the business	PO1.4 IT Strategic Plan PO4.2 IT Strategy Committee PO4.15 Relationships DS13.1 Operations Procedures and Instructions ME4.1 Establishment of an IT Governance Framework

7. References

[1] IT ASSURANCE GUIDE:
https://csbweb01.uncw.edu/people/ivancevichd/classes/MSA%20516/Extra%20Readings%20on%20Topics/COBiT/IT%20Assurance%20Guide%20Using%20COBit.pdf

[2] SAMPLE PROJECT 2
http://www.drkodali.info/isa652/SampleProject2.pdf

[3] SAMPLE PROJECT 1
http://www.drkodali.info/isa652/SampleProject1.pdf

[4] http://www.drkodali.info/isa652/