Securing Data in The Cloud Report

Carroll Reed, Collin Lightfoot, Michael Maizel, Norman McKinney, & Patrick Robertson

CST 620 Prevention of Cyber Attack Methodology

Dr. Cheryl Hinds

University of Maryland University College

Wednesday, November 14, 2018

Table of Contents

[Introduction 6](#_Toc530946918)

[Concepts and Scope 6](#_Toc530946919)

[**Cloud Computing Software Development Life Cycle** 6](#_Toc530946920)

[*Product backlog* 7](#_Toc530946921)

[*Sprint planning and sprint backlog* 7](#_Toc530946922)

[*Sprint execution and daily scrum* 8](#_Toc530946923)

[*Sprint review* 8](#_Toc530946924)

[*Sprint retrospective* 8](#_Toc530946925)

[*Next iteration* 8](#_Toc530946926)

[**Rationalization** 9](#_Toc530946927)

[*Agile Scrum life cycles* 9](#_Toc530946928)

[*Test-driven development* 9](#_Toc530946929)

[**Necessary Products** 10](#_Toc530946930)

[*Version control software.* 10](#_Toc530946931)

[*Agile project management software.* 10](#_Toc530946932)

[*Continuous testing software.* 10](#_Toc530946933)

[**Mission of Superior Banking** 11](#_Toc530946934)

[**Growing Need for Cloud Computing** 11](#_Toc530946935)

[**Scope of Security Architecture** 12](#_Toc530946936)

[*Security objective* 12](#_Toc530946937)

[*Dedicated firewalls* 13](#_Toc530946938)

[*Intrusion detection system (IDS)* 13](#_Toc530946939)

[*Vulnerability assessment (VA)* 13](#_Toc530946940)

[*Web application firewall (WAF)* 14](#_Toc530946941)

[*Load balancers* 14](#_Toc530946942)

[*Log management* 14](#_Toc530946943)

[**Risk Management Strategy** 15](#_Toc530946944)

[*Possible threats* 15](#_Toc530946945)

[*Possible vulnerabilities* 15](#_Toc530946946)

[*Likelihood of threats exploiting vulnerabilities* 16](#_Toc530946947)

[**Threat Modeling** 16](#_Toc530946948)

[Functional Design 17](#_Toc530946949)

[**Software Quality Requirement Engineering (SQUARE)** 17](#_Toc530946950)

[**SQUARE** 18](#_Toc530946951)

[*SQUARE Steps* 18](#_Toc530946952)

[Evaluation of Technology 22](#_Toc530946953)

[**Security Issues in Software System Environments** 22](#_Toc530946954)

[*Stand-alone Computing* 22](#_Toc530946955)

[*Client-server Model Computing* 22](#_Toc530946956)

[*Distributed Computing Model* 23](#_Toc530946957)

[*Cloud Computing Model* 24](#_Toc530946958)

[**Security Issues in Development Environments** 25](#_Toc530946959)

[*Traditional Waterfall Model* 25](#_Toc530946960)

[*Spiral Model* 26](#_Toc530946961)

[*Agile/Extreme Programming Model* 26](#_Toc530946962)

[*Cloud-based Model* 27](#_Toc530946963)

[*Cloud Provider Assurances* 27](#_Toc530946964)

[*Cloud Provider Confidentiality* 29](#_Toc530946965)

[*Cloud Provider Security Policy* 30](#_Toc530946966)

[**Analysis & Planning For Evaluating Technologies** 31](#_Toc530946967)

[*Role-based Access Control (RBAC)* 32](#_Toc530946968)

[*Mandatory Access Control* 33](#_Toc530946969)

[*Attribute-based Access Control (ABAC)* 33](#_Toc530946970)

[*Discretionary Access Control (DAC)* 33](#_Toc530946971)

[System Design Specifications 34](#_Toc530946972)

[**Design Specifications for Data-in-Transit Protection Model** 34](#_Toc530946973)

[Software Development Plan 36](#_Toc530946974)

[**Need for Software Development** 36](#_Toc530946975)

[**Plan for Software Development** 36](#_Toc530946976)

[*Beginning planning and design phases* 36](#_Toc530946977)

[*Sprint cycles* 37](#_Toc530946978)

[**Design and Development Considerations for the System** 37](#_Toc530946979)

[*Scalability* 38](#_Toc530946980)

[*Availability* 38](#_Toc530946981)

[*Manageability* 39](#_Toc530946982)

[Testing and Integration 39](#_Toc530946983)

[**Methods for Protecting Data** 39](#_Toc530946984)

[**Methods for Protecting the Integrity of Data** 40](#_Toc530946985)

[**Test Plan** 41](#_Toc530946986)

[*Analyze the product* 42](#_Toc530946987)

[*Design the Test Strategy* 42](#_Toc530946988)

[*Define Test Objectives* 43](#_Toc530946989)

[*Define Test Criteria* 43](#_Toc530946990)

[*Resource Planning* 44](#_Toc530946991)

[*Plan Test Environment* 45](#_Toc530946992)

[*Schedule & Estimation* 45](#_Toc530946993)

[*Determine Test Deliverables* 46](#_Toc530946994)

[**Compilation in Transit** 47](#_Toc530946995)

[Deployment Strategy 47](#_Toc530946996)

[**Tailoring of Data Protections for the SaaS Cloud Infrastructure** 47](#_Toc530946997)

[**Security for Cloud Computing** 48](#_Toc530946998)

[*Access Control* 48](#_Toc530946999)

[*Data at Rest* 49](#_Toc530947000)

[*Data in Use* 50](#_Toc530947001)

[*Data in Motion* 50](#_Toc530947002)

[Operations and Maintenance 51](#_Toc530947003)

[**Evaluate Business Activities** 51](#_Toc530947004)

[*Identify Critical Business Processes* 52](#_Toc530947005)

[*Label Dependencies* 52](#_Toc530947006)

[*Define Vital Applications* 52](#_Toc530947007)

[*Assess Current Data Recovery* 53](#_Toc530947008)

[**Determine Recovery Time Requirements** 53](#_Toc530947009)

[*Perform a Business Impact Analysis (BIA)* 53](#_Toc530947010)

[*Define Recovery Point Objectives (RPO)* 54](#_Toc530947011)

[*Distinguish Recovery Time Objectives (RTO)* 55](#_Toc530947012)

[*Determine Work Recovery Time (WRT)* 55](#_Toc530947013)

[*Designate Maximum Tolerable Downtime (MTD)* 55](#_Toc530947014)

[**Test Hypothesis** 56](#_Toc530947015)

[*Assess Risk* 57](#_Toc530947016)

[*Test Theory* 58](#_Toc530947017)

[*Redesign Accordingly* 58](#_Toc530947018)

[*Implement Solutions* 58](#_Toc530947019)

[**Policies & Procedures** 59](#_Toc530947020)

[*Develop Procedures* 59](#_Toc530947021)

[*Align Procedures* 59](#_Toc530947022)

[*Form a Team* 59](#_Toc530947023)

[**Auditing Plan for Security** 60](#_Toc530947024)

[Disposal Plan 61](#_Toc530947025)

[**Technology Disposal Concerns** 62](#_Toc530947026)

[*Avoiding Data Breaches* 62](#_Toc530947027)

[*Effective Processes, Policies and Procedures* 62](#_Toc530947028)

[*Outsources Technology Disposal* 63](#_Toc530947029)

[**Technology Disposal Options** 64](#_Toc530947030)

[Conclusion 65](#_Toc530947031)

[References 66](#_Toc530947032)

Securing Data in The Cloud Report

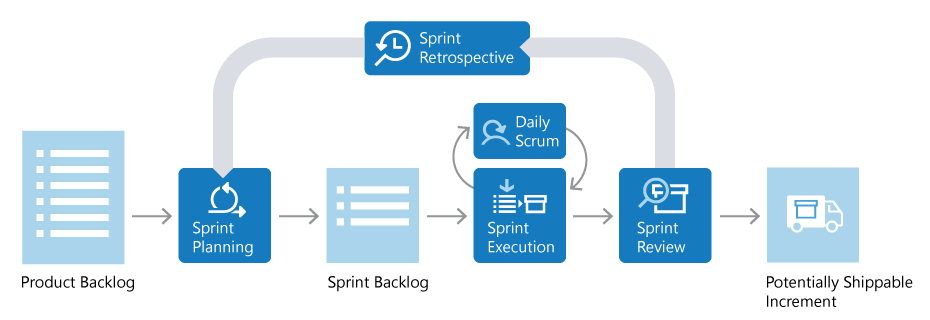
Carroll Reed, Collin Lightfoot, Michael Maizel, Norman McKinney, & Patrick Robertson

# Introduction

Superior Banking is moving to cloud computing to remotely handle all its online transactions to lower the expenditures on managing in-house server rooms. We were tasked to develop a software development life cycle (SDLC) report for the bank’s cloud computing architect who is currently designing the new cloud computing environment. This report is intended to both explain which SDLC to implement and the rationale behind the decision as well as define the data protection techniques necessary for a robust cloud-based service. With the information provided in this report, the architect should be able to implement a cloud computing architecture that will carry out the day-to-day banking operations Superior Banking provides while concurrently ensuring all data regarding transactions are safe and secure from any threats.

# Concepts and Scope

## **Cloud Computing Software Development Life Cycle**



***Figure 1: Agile Scrum-based SDLC* to be followed. Reprinted from Microsoft, by Gregg Boer (2017)**

Agile Scrum-based SDLCs, Figure 1, illustrates a design to take quick, incremental steps, known as sprints, towards developing a product that gains additional features at the end of each sprint. Superior Banking’s cloud architecture lifecycle development will follow a test-driven development approach where tests are created before development occurs.

### *Product backlog*

The product backlog is a list of deliverables the development team can deliver over a period. The items at the top of the backlog are usually the next items to implement and should, therefore, be implementable at the time their respective sprints begin.

### *Sprint planning and sprint backlog*

During the sprint planning phase, team members will choose which backlog items will be worked on during the upcoming sprint. These items are usually selected based on which ones have the highest priority as well as what the team itself believes can be completed within the forthcoming sprint before being placed in the sprint backlog. To allow for completion, each backlogged item is usually broken down into several tasks. The current sprint’s backlog items and their respective tasks are then listed on a task board that segregates the tasks in three sections: To Do, In Progress, and Completed. Team members will move these tasks to their appropriate sections during the sprint’s execution.

### *Sprint execution and daily scrum*

Once the sprint backlog has been filled with items and tasks to complete, team members choose amongst themselves which tasks they will each complete before executing the sprint. The team will have a daily meeting limited to 15 minutes where each member reports on the progress made since the previous daily meeting, what will be worked on today, and anything obstacles they have come across during the current sprint’s execution. Before any development can begin, each team member must create test cases that capture the requirement(s) their assigned tasks are trying to accomplish, as this SDLC follows a test-driven development approach and therefore requires test cases to be created before any tasked-specific functionality can be developed. Once the test cases have been built, team members will then complete their tasks be ensuring the defined test cases pass, thereby meeting the requirements said tasks are fulfilling.

### *Sprint review*

The team demonstrates what was accomplished during the current sprint.

### *Sprint retrospective*

During the retrospective, the team discusses its members’ performances during the latest sprint, with the outcome of the retrospective used as learning experiences for team members to take note of instead of the next sprint cycle.

### *Next iteration*

The entire cycle is repeated for the next sprint until all the items in the product backlog are completed and the final deliverable has been shipped for quality assurance testing.

## **Rationalization**

### *Agile Scrum life cycles*

Shorter, iterative cycles provide the development team with frequent opportunities to learn and improve from previous sprints. The iterative releases also allow for quick and responsive feedback from Superior Bank which helps adjust requirements necessary to meet the transactional needs Superior Bank requires, creating a better deliverable in the long run.

### *Test-driven development*

Defining automated test cases based from known requirements allows developers to first create minimum viable products (MVP) that meet their initial test cases (and therefore requirements) which can then be expanded to accommodate all the product’s intended sprint items’ requirements without having to backtrack as much if more requirements are discovered or created in future sprints. For example, having secure banking is necessary for any online transactions; developers should create test cases to ensure any item within the sprint adheres to the highest security standards online banking software should possess before creating test cases that will prove the MVPs have been created. Such developmental planning will ensure all the current sprint’s needs had been met and serve as a guide for developers on how to further develop the product should more requirements and tasks be created or discovered for the product after Superior Banking provides customer feedback.

## **Necessary Products**

### *Version control software.*

Version control software keeps track of every modification to a project within a centralized repository accessible to all team members. Without version control, the team cannot easily backtrack to a previous version of the project in case mistakes are made, requirements had changed, or requirements had been added. Some version control software also allows team members to be working on different tasks in similar areas of the project without causing any disruption in development or testing.

### *Agile project management software*

Agile project management software(APMS) grants team members a webpage where they can assign and manage tasks from the task board themselves, report any issues that arise during development or testing, and have forums for discussions during a sprint cycle. Most of these are free, integrate well with version control software, and are easy for a team to set up and use for almost any project.

### *Continuous testing software.*

Test cases might start to fail as the project has functionality added to it following the completion of each task, but having developers run the project’s automated test cases themselves prevents the developers from efficiently and quickly completing their respective tasks in each sprint: the sole reason why the Agile Scrum life cycle is used. The automated testing software is therefore required to build and test the software while developers focus on adding their tasks’ required functionality to the software before the sprint ends on separate machines.

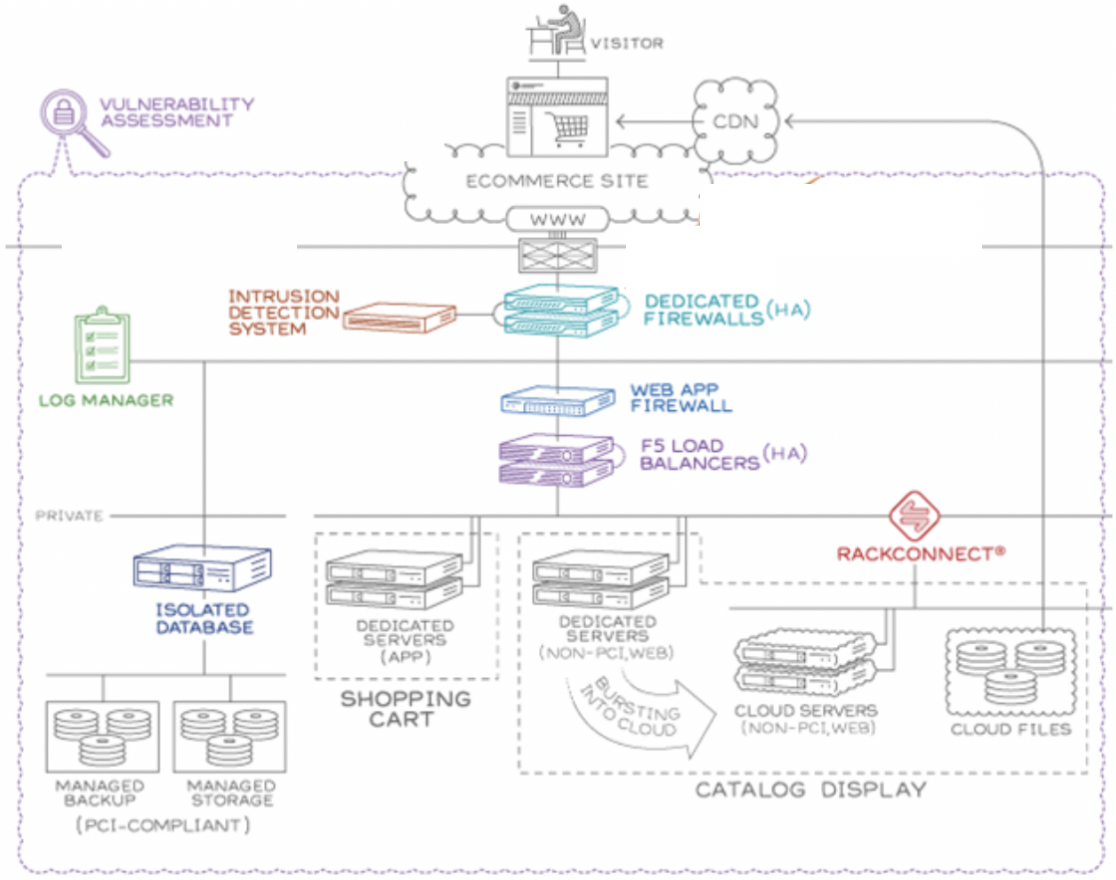
## **The mission of Superior Banking**

Superior Banking’s mission is “To serve the real estate, business, and financial needs of customers both online and in-store with a commitment to excellent service, profitability and sustained growth.”

## **Growing Need for Cloud Computing**

Cloud computing solutions have been rapidly increasing in demand and show no signs of stopping, as the cloud market is estimated to be worth $411 billion by 2020 (Networkers, 2018). Cloud computing solutions lead to company-wide savings, as businesses no longer maintain a staffed in-house server room and can instead divert fewer funds towards renting servers in a fully staffed and updated warehouse. Superior Bank also believes migrating to cloud computing solutions is a cheaper alternative to the server rooms it currently operates to manage its online banking services, which is the reason as to why the bank is migrating to the cloud.

## **The scope of Security Architecture**



***Figure 2*. *Superior Banking’s proposed cloud environment architecture topology.* Reprinted from Rackspace, Rackspace US, Inc., 2013.**

### *Security objective*

The architecture must protect all data in transit, as well as all data at rest. Data in transit, in this case, refers to customer online banking transactions, as having customer information stolen while the transactions occur will lead to financially severe damages of which Superior Bank would be liable. Data at rest pertains to all customer information the bank possesses, as having data stolen from the bank’s databases will also lead to grave financial damages of which the bank would be liable. Moreover, if any data is taken, then the bank’s reputation will most likely be damaged once its customers find out.

### *Dedicated firewalls*

A firewall is an application’s first line of defense; it serves as a keystone for all other security technologies involved. Placing firewalls between the Internet and the bank’s cloud infrastructure allows the bank to control traffic between trusted and untrusted networks.

### *Intrusion detection system (IDS)*

Attacks are bound to continually occur, and usually without the knowledge of the system owners. It is therefore essential to have controls, such as IDS’, in place that can detect when these attacks are happening for the system owners to quickly respond to the situation and minimize the possibility of data loss or systems compromised.

### *Vulnerability assessment (VA)*

A vulnerability assessment (VA) is a fundamental security requirement that helps identify any exploitable weaknesses in a cloud infrastructure. Periodic, weekly system scans are necessary to ensure the timely identification of vulnerabilities that can be introduced after changing configurations or a newly discovered vulnerability. VA’s are proactive and preventive controls that help identify an infrastructure’s weaknesses before they are exploited. Attackers use scanning systems to find weaknesses to exploit, so the bank must act, scan, and repair system weaknesses before they can be exploited.

### *Web application firewall (WAF)*

A web application firewall (WAF) filters and blocks non-essential traffic to a system’s application layer to try and protect insecure applications from exploits: in general, a network firewall cannot prevent these types of attacks by itself. One such protection provided by WAFs are blocking SQL injection attempts. Applications are the third most-used attack vectors while web application vulnerabilities are responsible for over a third of total data loss among leading companies, hence why WAFs are critical to any cloud infrastructure (Verizon, 2018).

### *Load balancers*

Load balancers provide several security features, including the ability to disable SSL traffic, centralized certificate management, HTTP and HTTPS session persistence, and central restriction of weak SSL ciphers. Load balancers also allow servers to be taken out of the running environment to be updated with the latest firmware and software patches while giving more work for the other servers in the meantime. Load balancers also normalize TCP traffic, which helps protect against insertion and evasion network attacks (attackers use these to counteract or evade IDSs and firewalls).

### *Log management*

Log management helps detect, protect and respond to security incidents. Log management solutions protect the infrastructure by identifying unauthorized access attempts before any sensitive customer data is customized, detect unauthorized changes and misuse of privileges within the cloud architecture, and helps respond to attacks and shorten recovery times by providing descriptive log data which can support breach investigations.

## **Risk Management Strategy**

### *Possible threats*

External threats, such as hackers and organized crime groups, may attempt to infiltrate Superior Bank’s databases by executing potential SQL injections on the databases’ web applications access points or by embedding malicious code on trusted websites accessible by computers within the banks network known as cross-site scripting attacks. Insider threats can also exist and wreak havoc. Disgruntled or dishonorable employees could use the internet to smuggle information in and out of the clinic by sharing the clinic’s data with unauthorized outside parties.

Other employees may leave workstations, printed documents and other data accessible or viewable to unauthorized parties. Employees may even use their workstations to browse websites not related to work, which may cause the web security threats above to occur. Staff also may neglect to install vendor-supplied updates and patches to the software or firmware being used, allowing the system to remain vulnerable by these patchable or updatable attack vectors. The staff could also connect infected USB devices to the system’s workstations, or computers to the system’s ethernet lines, and unknowingly install malicious software into the workstations or the system itself. Employees could also have generic passwords that can be quickly cracked or may share their respective passwords with other employees.

### *Possible vulnerabilities*

Software or firmware could be outdated, physically accessible to unauthorized users, or inefficient or ineffective against certain attacks. Poorly trained employees could also lead to security leaks or an improperly configured security infrastructure.

### *Likelihood of threats exploiting vulnerabilities*

With Verizon (2018) alone finding over 53,000 cyber-attack incidents that created 2,200 successful breaches, Superior Bank has a high probability of being one of the thousands of companies attacked.

## **Threat Modeling**

One must look at the likelihood of a threat exploiting a vulnerability as well as the level of damage the exploit could bring to the targeted system. To do this, we recommend using the Risk Rating Scale below to determine which risks should be mitigated first.

***Table 1: Risk Rating Scale***

|  |  |  |  |
| --- | --- | --- | --- |
| Risk Rating Scale | | | |
| Likelihood | **Severity** | | |
| **Slightly Harmful/Low** | **Harmful/Medium** | **Extremely Harmful/High** |
| Highly Unlikely/Low | Low Risk | Medium Risk | Medium Risk |
| Unlikely/Medium | Medium Risk | Medium Risk | High Risk |
| Likely/High | Medium Risk | High Risk | High Risk |

# Functional Design

## **Software Quality Requirement Engineering (SQUARE)**

Part of the SDLC is requirements gathering and analysis. If done improperly, this process will lead to software projects being late, over budget or outright canceled due to potential cost overruns while trying to improve security concerns of applications already fielded. When security analysis and security-oriented engineering is implemented early in the software development cycle, studies found a 12 to 21 percent return on investment. As can be seen from Figure 1, the cost to fix requirement problems identified later in the project costs close to $2 million.

***Table 2: Cost of Fixing Requirement***

|  |  |  |  |
| --- | --- | --- | --- |
| *Cost of Fixing Requirement* | | | |
| Stage | **Critical Bugs Identified** | **Cost of Fixing One Bug** | **Cost of Fixing All Bugs** |
| Requirements |  | $139 |  |
| Design |  | $455 |  |
| Coding |  | $977 |  |
| Testing | 50 | $7,136 | $356,800 |
| Maintenance | 150 | $14,102 | $2115,300 |
| Total | **200** | **$22,809** | **$2,472,100** |

***Table 3: Cost of Fixing Requirement Links***

|  |  |  |  |
| --- | --- | --- | --- |
| *Cost of Fixing Requirement* | | | |
| Stage | **Critical Bugs Identified** | **Cost of Fixing One Bug** | **Cost of Fixing All Bugs** |
| Requirements |  | $139 |  |
| Design |  | $455 |  |
| Coding | 150 | $977 | $146,550 |
| Testing | 50 | $7,136 | $356,800 |
| Maintenance |  | $14,102 |  |
| Total | **200** | **$22,809** | **$503,350** |

## **SQUARE**

One of the methodologies that can be used to ensure security is factored in while requirements are gathered and analyzed is Software Quality Requirements Engineering (SQUARE). SQUARE is a stepwise methodology for eliciting, categorizing and prioritizing security requirements for information technology systems and applications. In the early stages of the SDLC, using SQUARE ensures security requirements are analyzed along with the system functionality requirements. SQUARE is determined through a process of nine discrete steps.

### *SQUARE Steps*

* Agree on definitions - include all definitions that are relevant to the project, even if they are common terms.
* Identify assets and security goals - development teams need to formally agree on a set of prioritized security goals for the project. Without overall security goals for the project, it is impossible to identify the priority and relevance of any security and quality requirements that are generated.
* Develop artifacts to support security requirements definition - documented normal usage and threat scenarios, misuse cases, and other documents needed to support requirements definition.
* Assess risks - requires an expert in risk assessment methods, the support of the stakeholders, and the support of a requirements engineer.
* Select elicitation techniques - sitting down with a primary stakeholder to try to understand that stakeholder’s security requirements need.
* Elicit security requirements - these builds on the artifacts that were developed in earlier steps, such as misuse and abuse cases, attack trees, threats, and scenarios.
* Categorize requirements - allows the requirements engineer to distinguish among essential requirements, goals, and architectural constraints that may be present.
* Prioritize requirements - depends on the prior step and may involve performing a cost/benefit analysis to determine which security requirements have a high payoff relative to their cost.
* Inspect requirements - once the inspection is complete, the organization should have an initial set of prioritized security requirements.

***Flow Cart 1: SQUARE Steps***

***Table 4: The SQUARE Process (Mead, 2013)***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Input | Techniques | Participant | Output |
| Agree on definitions | Candidate definitions from IEEE and other standards | Structured interviews, focus group | Stakeholders, requirements team | Agreed-to definitions |
| Identify assets and security goals | Definitions, candidate goals, business drivers, policies and procedures, examples | Facilitated work session, surveys, interviews | Stakeholders, requirements engineer | Assets and goals |
| Develop artifacts to support security requirements definition | Potential artifacts (e.g., scenarios, misuse cases, templates, forms) | Work session | Requirements engineer | Needed artifacts: scenarios, misuse cases, models, templates, form |
| Perform risk assessment | Misuse cases, scenarios, security goal | Risk assessment method, analysis of anticipated risk against organizational risk tolerance, including threat analysis | Requirements engineer, risk expert, stakeholders | Risk assessment results |
| Select elicitation techniques | Goals, definitions, candidate techniques, expertise of stakeholders, organizational style, culture, level of security needed, cost benefit analysis, etc. | Work session | Requirements engineer | Selected elicitation techniques |
| Elicit security requirements | Artifacts, risk assessment results, selected techniques | Joint Application Development (JAD), interviews, surveys, model-based analysis, checklists, lists of reusable requirements types, document reviews | Stakeholders facilitated by requirements engineer | Initial cut at security requirements |
| Categorize requirements as to level (system, software, etc.) and whether they are requirements or other kinds of constraints | Initial requirements, architecture | Work session using a standard set of categories | Requirements engineer, other specialists as needed | Categorized requirements |
| Prioritize requirements | Categorized requirements and risk assessment result | Prioritization methods such as Triage, Win-Win | Stakeholders facilitated by requirements engineer | Prioritized requirements |
| Requirements inspection | Prioritized requirements, candidate formal inspection technique | Inspection methods such as Fagan, peer reviews | Inspection team | Initial selected requirements, documentation of decision-making process and rationale |

# Evaluation of Technology

## **Security Issues in Software System Environments**

### *Stand-alone Computing*

Stand-alone computing is a self-contained environment with one or more terminals for one or more concurrent local user. It is one of the most straightforward computing architecture environments for use and security because its boundaries are well-established and because one can physically and logically monitor any interactions with its data, applications or hardware environment. However, stand-alone computing has little use in a world that requires multiple, diverse data sources to be connected via a network to perform application processing and analysis.

### *Client-server Model Computing*

Developers eventually realized that hosting data storage and processing requirements onto one or more separate computing systems, such as servers, allows a user’s machine to run faster, leading to the creation of the client-server model. Initially, a single server hosted many applications that provided computationally-intense services for programs on a user’s machine. This model eventually expanded where larger applications and data sets are now usually hosted on their machines to provide their respective software application services.

Clients and servers were first connected with data lines, with amplifiers and repeaters extended the range of the connections over longer distances. Much other networking hardware and software was created to allow most computers the ability to connect remotely, forming the basis of the Internet we have today. Managing which systems communicate with one another and how is the leading security challenge client-server models pose to local networks and their machines.

### *Distributed Computing Model*

Once networking computers became popular, system engineers started designing servers to take on more work within a network, popularizing the idea of using servers for processing- and data-intensive applications: applications were distributed over multiple servers in warehouses with many stacks of servers known as server farms or throughout numerous physical locations. Businesses soon realized the importance of backup systems in case the central server systems went down, leading to more servers being housed in even more physical locations.

One such use of distributed contributing is dividing up a large and complex problem into many significantly smaller parts which are then distributed to the many servers to quickly solve and return the answers to those pieces that are eventually concatenated and used to calculate the final solution: a process known as the “divide and conquer” approach. Multiple servers are also used for computers to communicate across the Internet with one another.

### *Cloud Computing Model*

Cloud-computing models have more intensive distributed processing designs, as their data processing infrastructure components, servers, applications, and networks, not only run in-parallel but are distributed globally and even across services providers. This cloud computing environments are generally owned by ISPs and contracted out to companies after signing service-level agreements. The data processing cloud resources depend on several criteria, including how fast and how many processors are used as well as the speed of network connections between cloud resources and the Internet.

There are also three types of cloud architectures and cloud resources: private, public, and hybrid clouds. The private cloud is contained within the boundaries of a corporation or individual residence, while a public cloud shares resources available for the public to include and process data, and a hybrid cloud uses a mix of public and private clouds to provide solutions for users. There is also a government cloud, which is a private cloud architecture owned by one or more government organizations.

Most clouds are composed of at least five components: the consumer, the provider, the auditor, the broker, and the carrier. The consumer is the body that maintains a business relationship with the cloud’s providers to use their services. The provider is an entity responsible for making a service available to consumers. The auditor conducts independent assessments of cloud services, operations, performances, and security of the cloud architecture.

The broker manages the use, performance, and delivery of the cloud services and negotiates the business relationships between the cloud’s consumers and providers. The carrier is the intermediary that provides the connectivity and transport of cloud services from the providers to the consumers. If all components work in tandem, then the cloud’s services function smoothly. Like the client-server and distributed computing models, controlling network traffic and server or warehouse malfunctions and shutdowns are significant security challenges posed by this model. Public cloud service providers also need to worry about the type of traffic channeled through the cloud architecture in case the malicious software is being run in the cloud solutions either on purpose or accident.

## **Security Issues in Development Environments**

### *Traditional Waterfall Model*

The waterfall development model was the first process model to be introduced and is easy to both understand and use. The whole development process is divided into separate phases: requirement gathering and analysis, system design, implementation, testing and integration, deployment, and operations and maintenance. These phases cascade to each other in a process that only flows steadily downward to the next phase, making the design look like a waterfall.

These phases usually are clearly defined and allow for a comfortable task arrangement, the model produces deliverables late through the process and does not adapt well to any new requirements or technological developments while under development (TutorialsPoint.com, n.d.). Therefore, for complex projects that have a high risk of changing needs, such as security-related projects, developers do not possess the means to adopt new security requirements if following this model.

### *Spiral Model*

Spiral models follow the waterfall model’s cascading process of development while allowing for multiple passes through the waterfall model’s phases. Each pass follows specific prioritized design and development criteria to create the overall system incrementally. Performing a risk analysis at the beginning of each phase allows for developers to focus on mitigating any potential risks as well as adapt their project to meet any newly-defined requirements when determining what design and development criteria to prioritize. This model heavily relies on the risk analysis phase, meaning mistakes made in this will ripple throughout the project and most likely cause the project not to meet specified requirements during a pass and ultimately fail (IS2200, 2013). Therefore, this model is not appropriate for high risk, security-based projects.

### *Agile/Extreme Programming Model*

The Agile design model is one category of the extreme programming models that has grown in popularity amongst most engineers and developers. The rise in popularity is due to being team-focused when it comes to task prioritization as it allows for the rapid development of prototype models and components of a system to-be-developed. Agile passes are typically done in two- to three-week increments with some Agile teams having informal stand-up meetings daily or weekly to assess what current team members have and are working on during an increment.

Agile processes usually have fewer requirements documentation, plans, and procedures than waterfall or spiral models due to the swiftness of development that is necessary for each increment. The lack of focus for documentation allows for Agile teams to focus more on capturing any design issues that arise, or requirements that are added, during each increment giving the overall process greater flexibility than waterfall or spiral models. The high degree of flexibility and the rapid prototyping and testing are the main reasons why most security-based projects should use this model.

### *Cloud-based Model*

Cloud models usually involve integrated development environments, application lifecycle management components, and application security testing components hosted on remote machines. Lifecycle management components include test and quality management, source code and configuration management, and continuous delivery tools. Besides using remote computers for automated tasks and storage, cloud-based models are mainly used for rapid prototyping, parallel branch development and load- and performance testing.

Developing teams are starting to switch to this development model over Agile due to most of the computational load being shifted to remote machines to allow for quicker development and prototyping (Violino, 2011). The automated security and testing components ensure that all deliverables maintain their defined requirements throughout the developing process, which further promotes why this is quickly gaining popularity for quacking-changing, high risk, ongoing projects such as ones focusing on application security.

### *Cloud Provider Assurances*

1. ***How could a cloud-based solution maintain a proper authentication system for its clients?*** Besides keeping the authentication systems up and running, appropriate client authentication systems should continually identify and authenticate users, assign user access rights, and create and enforce resource access policies. Multifactor authentication should be used to both identify and authenticate users to reduce the risk of credentials being stolen or compromised as attackers will need to acquire multiple, independent authentication elements to compromise a client’s credentials successfully. Also, Roles should be designed to adequately distribute user access rights out to a client’s users to ensure no one person can adversely affect the entire cloud solutions environment.

Moreover, cloud service providers frequently offer several different types of storage devices for each of their services. Consumers should be able to somewhat easily create and enforce unique access policies to their storage devices so that only certain people under certain conditions can access the protected data (Faatz, 2018). With these factors addressed, cloud-based solutions can maintain proper client authentication systems for their consumers.

1. ***How might a cloud-based solution ensure that one client's data is kept confidential and protected from other clients who also have access to the same data center?*** Cloud-based solutions should have server connection protocols that only permit the renters of the servers to access said servers remotely, as well as enforcing a one-client-per-server policy, as these policies will ensure only the clients (and those they grant server access to) can access the clients’ data. The confidentiality and protection of client data. Client segregation and control policies are the easiest ways to ensure a client cannot access another client’s data.
2. ***What type of assurances would a client expect that the security of the software components and utilities provided by the cloud-based solution provider will be consistently maintained if the distributed systems are owned and leased from other organizations?*** Clients would expect data stored on their distributed systems to be inaccessible to unauthorized users as well as having their distributed systems protected from known cyber-attacks. Cloud services providers should also be able to grant clients the ability to see who is using their services for any illegal activities, so the clients themselves are not held responsible for any malicious actions.

### *Cloud Provider Confidentiality*

1. ***If we had a company that processed and kept medical imaging data, how might a corporation called Medical Imaging (MI) keep other cloud subscribers from accessing MI's data?*** Using multifactor authentication and the principle of least privilege through the assignment of roles. Multifactor authentication will force other cloud subscribers to acquire multiple, independent identification factors before they can attempt to successfully compromise the MI company’s user access credentials and access its medical imaging data. Having the MI company assign roles to its users where each role only has access to the medical imaging data necessary to fulfill their respective role’s duties will ensure that a client’s whole system is not accessible if a user’s account is compromised.
2. ***How could the Medical Imaging corporation manage the images split across multiple third-party ISPs?*** Having images split across multiple third-party ISPs means that the MI company has its data stored in various locations owned by, but a part of, the company’s cloud services provider. Documents spread across various sites means the company needs to ensure these remote locations have the necessary security services defined in the previous question on each of the remote location’s machines for the images to retain their confidentiality.

### *Cloud Provider Security Policy*

1. ***How can another corporation ensure a similar level of security as the cloud-based ISP provider for MI?*** To provide a similar security level as the cloud-based ISP for the MI company, another corporation must incorporate the security features and assurances listed under the Cloud Provider Confidentiality section of this report. The NIST 800 and 1800 Special Series suggest assurances and services such as privileged account management, multi-factor authentication, border gateway protocol route origin validation and derived personal identity verification (NIST, 2018).
2. ***How could the security policy defined by the cloud-based ISP provider be maintained and ensured at the application level? What types of agreements would be needed? What types of software, systems, and security testing would you require?*** Application-level security policies should require multi-factor authentication for users to force attackers to obtain multiple different identification elements to compromise a single user’s credentials and thus maintain and ensure a high level of security. Service-level agreements that detail such security policies would be needed, and regularly-scheduled vulnerability assessment scans would occur throughout the entire cloud infrastructure to ensure timely identifications of vulnerabilities that could have been introduced after the client changed configurations or after finding a system design flaw. Dedicated firewalls and web application firewalls would be some of the systems necessary to mitigate malicious traffic from entering the cloud systems while intrusion detection systems would attempt to detect any malicious system intrusion. Log management would help identify, protect and respond to security incidents while honeypots could be set up for the cloud system administrators to study malicious threats attempting to break into and attack the cloud infrastructure.

## **Analysis & Planning for Evaluating Technologies**

Server virtualization is the process of using a central physical server to support virtual servers and applications that are hosted over a remote connection. Companies have started to move to cloud computing to increase efficiency in the workforce and provide a cost savings (Rouse, 2009). Cloud servers can be configured with virtualization software such as VMware or Microsoft Virtual Server that allows for each virtual server to run multiple operating systems simultaneously.

Virtual environments such as this reduce capital expenditures by creating a software layer that simulants hardware and software for a user to interface. With the operating system being simulated, any custom configuration can be established at the cloud level without adjustments needed on the user’s terminal. Furthermore, the processing power of the user’s terminal does not affect the operating power of the virtual server since it is being shared with a large physical server that can prove its resources based on demand (Kissel et al., 2008).

The most significant benefit of using virtual servers is the sharing of resources. Sharing resources gives a company the ability to pay only for what they need. Having the ability to increase storage space or add components without investing in large capital servers give a company the ability to adjust quickly. With cloud computing, software patches and upgrades can be applied to all users immediately without having to push program installs to each terminal individually. Mobile Cloud Computing (MCC) continues to be a required feature for most industries.

The ability to gain access to programs and databases with smaller mobile devices allows for total portability and greater collaboration. Since the processing power of the virtual system is being handled at a central location, terminals do not require much to display the operating system or application they are running (Akherfi, Gerndt, & Harroud, 2018). This configuration allows for mobile devices such as smartphone and tablets to have nearly the same ability as a regular computer.

One of the notable drawbacks with smaller devices having access to full cloud resources is they are currently limited on storage and power thus limiting functional securely (Akherfi, Gerndt, & Harroud, 2018). Encryption on data being transmitted to and from the cloud protects against unauthorized access or interception. Since information used during cloud computing is stored remotely and transmitted over the internet, some encryption should be used depending on how sensitive the files are.

With the chance that user credentials could be stolen, giving authorized access to the virtual system, all sensitive data should maintain encryption while being stored on the cloud. Four types of access control that are highlighted are role-based access control (RBAC), mandatory access control (MAC), attribute-based access control (ABAC),and discretionary access control (DAC). The most robust control method for this situation would be the role-based access control. RBAC is simple, flexible and provides security. Other control methods are not as practical within the contents of our industry and give too much access.

### *Role-based Access Control (RBAC)*

RBAC is a control system that uses the user’s role to determine their access level and permission settings. This method simplifies access and permission setting for individual users that share the same role in a company. RBAC allows administrators to uniformly apply or grant access for a set of users across the entire network (Hu, Ferraiolo, & Kuhn, 2006).

### *Mandatory Access Control*

MAC is a control system that compares data security with the corresponding user security level. MAC is most frequently used in a government setting where data or access is classified into different levels of sensitivity labels (Hu, Ferraiolo, & Kuhn, 2006).

### *Attribute-based Access Control (ABAC)*

ABAC has similarities to MAC but can use different attributes to allow access. ABAC provides more dynamic access control since it is not based on the user’s access level alone (“Using Security Labels for Directory Access,” n.d.).

### *Discretionary Access Control (DAC)*

DAC is a less restrictive control method. This method allows users control over data they own and can assign permissions for other users. A concern with this method would be the lack of control and ability to remove access when the users no longer need. Extra users have access to files and data adds the risk of inadvertent exposure. (Hu, Ferraiolo, & Kuhn, 2006)

# System Design Specifications

## **Design Specifications for Data-in-Transit Protection Model**

Data must be encrypted when being transmitted across the cloud infrastructure to prevent eavesdropping data during transmit. Firstly, email is not considered secure and therefore must not be used to transmit any sensitive data throughout the cloud systems. If sensitive data needs to be sent over email, then the email itself must be secured using cryptographically strong email encryption tools such as PGP encryption so only the intended recipient can easily decrypt the message.

If the email needs attachment(s) attached to it, then a file encryption tool should be used to encrypt the file(s) before attaching it or them to the email. Secondly, any data intended for transit must be transmitted over a Secure Sockets Layer (SSL) using the Transport Layer Security (TLS) protocol. SSL and TLS provide privacy and data integrity between two communicating applications and is designed to encapsulate other protocols such as HTTP (FISMApedia, 2008).

Non-web transmission of sensitive data should be encrypted with application-level encryption. Moreover, since the application database resides outside of the application server in the case of our cloud infrastructure design, we must create an encrypted connection using FIPS compliant cryptographic algorithms, like AES, between the database and the application to protect from outside surveillance.

Network-level encryption, such as IPSec or SSH tunneling should be used where application-level encryption is unavailable for non-web covered data traffic (UC Berkeley, 2018). In general, all transmitted, sensitive data should be encrypted in protected subnets with strong firewall controls. Below is a table of secure network protocols to use based on how a system wants to pass data throughout the network. *Table 1: Network Protocol To Use* describe which network protocol to use based on the current intended function of a machine.

***Table 5: Network Protocol to Use***

|  |  |
| --- | --- |
| Function | Network Protocol to Use |
| Web Access | HTTPS |
| File Transfer | FTPS, SFTP, SCP, WebDAV over HTTPS |
| Remote Shell | SSH2 Terminal |
| Remote Desktop | radmin, RDP |

When picking an encryption algorithm to use, make sure the key length is at least 1028 bits to provide stronger protection and that long and complex passphrases are used. Also, note that stronger encryption techniques will consume more CPU resources than a weaker encryption technique on average. Strong encryption algorithms include AES, Twofish, and RSA; each algorithm is useful for different purposes and data can be encrypted with multiple encryption algorithms at the same time to make data even more secure. AES is good to use when keys are being shared amongst multiple users, Twofish works will focus on smaller CPUs and hardware, and RSA should be used when data is supposed to be sent to only a single recipient (Kryptotel, n.d.). Moreover, WPA3 should be used for any wireless connections, as all other wireless encryption standards are insecure.

# Software Development Plan

## **Need for Software Development**

To maximize efficiency, most software projects, whatever the size, require at least some form of planning and designing before any code can be written. The planning and designing steps should then be divided into separate, distinct phases for the project’s coders to coherently determine such things as project requirements and the overall project’s design. These phases allow everyone involved to have a similar idea on how the project should turn out based on what requirements, as well as other considerations, are currently known.

The planning and designing phases, along with the implementation and testing phases, are ordered into specific patterns called software development processes or life cycles. Without these phases being both clearly defined and ordered, a coder can quickly become disorganized during a project’s implementation phase. Therefore, software development is a necessary component to any project’s development, as it helps the team focus on an overall design as well as manage the project’s implementation based on the set of agreed-upon requirements and other considerations that act as guidelines.

## **Plan for Software Development**

### *Beginning planning and design phases*

We must first go over what Superior Bank wants its cloud infrastructure to be able to handle when it comes to business transactions and create a set of requirements based off the functionality Superior Bank wants its cloud infrastructure to have. With these requirements, we can begin going over how we might design the entire cloud infrastructure; the infrastructure will most likely be broken up into separate components for us to develop and focus on incrementally. Next, we will establish sets of tasks that meet the project’s requirements, with each set gradually building of the previous set or adding new functionality entirely and place these tasks within our product backlog.

### *Sprint cycles*

With the sets of tasks generated, we can begin planning which set(s) of tasks from our product backlog we will work on during the first sprint cycle. The team will then complete these sets of tasks during the first sprint while at the same time having daily meetings to discuss the progress each of us has made on our respective tasks. Each task will first require team members to create test cases that prove each task’s requirements have been met once the task is completed before any implementation can begin.

A sprint review is conducted at the end of the sprint where the team discusses its members’ performances during the latest sprint, with the outcome of the retrospective being used as learning experiences for team members to take note of instead of the next sprint cycle. The next sprint’s backlog is then filled with the tasks remaining in the product backlog with more sprints following in the same way until all the tasks in the product backlog have been completed, therefore meeting all Superior Bank’s requirements with an acceptable cloud infrastructure.

## **Design and Development Considerations for the System**

All projects are required to be completed within a specific timeframe, a certain amount of human resources, and a specific amount of funding, which therefore limited to many constraints that lead to several design and development considerations. The following sections describe several design and development considerations Lydford (2015) states teams should have for cloud-based applications.

### *Scalability*

The team should consider if there should be requirements that handle any increases in load and demand on any of the components composing the cloud infrastructure. Teams should mainly focus on each part individually as well as how to avoid any contention issues and bottlenecks that might occur within the infrastructure.

### *Availability*

The team should consider what parts of the application have the highest risks of failure as well as in which parts of the system these failures would have the most impact. The team should also denote which parts of the application can benefit from redundancy and failover options, as well as determining whether data replication services will be necessary and if corrupt data has the potential of being replicated. Moreover, in the event of a disaster, how will the cloud infrastructure recover, how much data is acceptable to lose, how will we handle losing messages and data currently being processed and do we have backups for at least some of the transaction data.

The team also needs to consider what the acceptable levels of the system’s performance will be, how performance will be measured, and what the system should do if performance falls below the desired level if traffic spikes, disasters, etc. occur. The team should also consider the laws in which the data is being held as well as who has access to the transactional data and how the team can restrict and control access to any of the cloud infrastructure’s components.

### *Manageability*

The team should also consider how they plan on monitoring all or some activities occurring within the cloud infrastructure and where the monitoring data will be stored within the system. The team must figure out how to access this data, as well as whether some of the monitoring data can be lost if something happens to the system. Finally, the team must consider how the cloud infrastructure deployment is automated, how the system can be patched and redeployed without disrupting the live system.

# Testing and Integration

## **Methods for Protecting Data**

Data is most vulnerable when it is continuously on the move or in transit and make it easier for hackers to steal information because they will use specific security tools, such as Wireshark, to steal data. One of the best ways in securing data in transit is making sure that multiple or individual workstations use a virtual private network (VPN) when communicating over an unsecured network. Virtual private networks (VPNs) are a type of technology that is used to provide the necessary security and encryption connection when communicating over an unsecured network.

There are types of VPNs that are commonly used within many companies, remote access VPNs and site-to-site VPNs. During the remote access VPN, a public telecommunication technology is used to provides a secure remote connection over the Internet to the company’s network. An employee will connect to a public Wi-Fi hotspot using a gateway to gain access to the Internet, which will allow them secured connection to the company’s network. During a site-to-site VPN, a gateway technology device is being used to connect a company’s network from a location to another location within a different network (Rouse & Burke, 2018).

## **Methods for Protecting the Integrity of Data**

Integrity is one of the security areas of the CIA—confidentiality, integrity, and availability— the triad of information, which deals with maintaining the information consistency, accuracy, and trustworthiness. One of the leading security technologies that are used when protecting the integrity of data is access controls (Rouse, Haughn, & Gibilisco, 2014). Access controls are security measurements that are used to determine the access into the computer system and its network. These types of security measures are critical in providing security to the integrity of data. Three basic access controls are commonly used in many companies that offer a different type of access control security:

* Mandatory Access Control (MAC)
* Discretionary Access Control (DAC)
* Role-Based Access Control (RBAC)

Mandatory Access Control (MAC) uses a predefined set of rules that are put in place to determine the personnel that allows access to the company’s resources. Most companies that use this type of access control are standard within the government because this access control provides much more restricted access to a company’s resources. Discretionary Access Control (DAC) is less restricted than MAC because it depends on the owner of the access control. The owner of this access control will set the rules to allow only certain personnel that he/she wants to access the company’s resources.

Role-Based Access Control (RBAC) uses a set of rules that only allow the user to access to company’s resources that are an essential role within the company. An employee could be assigned different roles within the company. This type of access control will grant the user access to those company’s resources that allow him/her to complete their given roles and responsibility.

## **Test Plan**

There are eight critical steps that we will use doing the testing the security of the cloud computing technology. *Flow Chart 2: Test Plan Process* address the different steps that are used during the testing plan of the new cloud computing technology. This plan will ensure that data is well secured throughout the system development life cycle (SDLC) (“How to Create a Test Plan,” 2018).

***Flow Chart 2: Test Plan Process***

### *Analyze the product*

The first step in our testing plan is to analyze the product because knowing about the technology that needs to be tested allows us to get a feel for implementing the new technology. The product and service that is being tested is cloud computing, which is being used to store sensitive employment information, such as personally identifiable information (PII) (“How to Create a Test Plan,” 2018).

### *Design the Test Strategy*

The next step in developing a test plan is developing a test strategy. *Flow Cart 3: Design Test Strategy Process* address the design of the test strategy approach. During this step, four different steps will be fellow when establishing a strategic approach to the step (“How to Create a Test Plan,” 2018).

* Define Scope of Testing: the components, such as the hardware and software, that will be tested during the testing process.
* Identify Testing Type: establish what is being tested and the testing method that will be used.
* Document Risks & Issues: establish a report on the different risks and issues that may surface throughout the testing phases.
* Create Test Logistics: establish and create a logic testing format that is used to determine the correct testing method.

***Flow Cart 3: Design Test Strategy Process***

### *Define Test Objectives*

During this part of the testing process, we must define the primary objective of the test, such as listing the software features that will be tested, such as the security and the protection of both the data itself, as well as the integrity of data. The main test objectives will be ensuring that data being protected when using Service as a Software (SaaS) cloud computing technology. We will test the security features and the goals that are designed to ensure that the personally identifiable information (PII) is protected during the transition process (“How to Create a Test Plan”, 2018).

### *Define Test Criteria*

The define test criteria step, we will run a what if logical, critical statement that will help to determine whether security is protected. The what if statement will allow us to work out all the bugs before releasing to the public or in this case Superior Banking before they adopt the new software into their network. We will continue to run the test until the percentage of security breaches is low to a reasonable rate that does not cause the company to go bankrupt or lose its reputation. This information will need to be documented into a test metric document that displays the run rate and the passing rate in reaching our primary goal of providing security to data in transit. Before exiting the criteria, we will make sure that the run rate is at 100% in making sure that all testes and its outcomes are correctly run (“How to Create a Test Plan”, 2018).

### *Resource Planning*

The resource planning part of the test plan is a detailed summary of all the resources that have been used to complete this project, from human interactions to software installation and testing. *Table 6: Human and Software Installation Setup* describe each component and the crucial roles that are used throughout the project (“How to Create a Test Plan”, 2018).

***Table 6: Human and Software Installation Setup***

|  |  |  |
| --- | --- | --- |
| *Human and Software Interaction* | | |
| Human Interaction | Test Manager | * overseas the project, guide the direction of the project * gather the necessary resources needed to complete the project * making sure that all areas are properly cared by the other members of the projects |
| Testers | Penetrating testers   * will identify and describe the testing tools and techniques that will be used to test different areas of security. * Verify and help with the testing approach * Execute, logging results, and report any vulnerabilities within the system |
| Developer testers   * Implementing test cases, test programs, testing software |
| Test Administrator   * Making sure the testing environment is managed, as well as managing and maintaining the test environment. * Support different testers in using the test environment when executing the test, such as on a virtual machine |
| Software Quality Assurance (SQA) Members | * Making sure that the quality of the product is assurance * Making sure that all testing process is meeting the requirement for the test |
| Software Interaction | Server | * Installing the proper software installation for the cloud |
| Testing tools | * Using different types of penetrating techniques and testing tools * These tools would include Wireshark, virtual box machine, and Hadoop cloud sandbox. |
| Network | * The network that would be needed is a local area network (LAN) and a virtual private network |
| Computer equipment | * The personal computer systems that will be connected to the cloud, as well as remotely. |

### *Plan Test Environment*

During this testing stage, we will establish an appropriate testing environment that will be used to determine a stable outcome. The cloud computing technology that will be testing will be based on a networking environment and remotely, since most of the company’s resources will be uploaded/downloaded remotely (“How to Create a Test Plan”, 2018).

### *Schedule & Estimation*

During this step, it’s essential to establish a set schedule and estimation of when each of the different testes will be performed, as well as the human resources hour that will be put into the test. The testing process needs to occur on a bi-monthly basis to ensure that all security areas are tested. Each member of the testing team will provide a certain amount of human resources hour that equals to 280 human resources hours of testing. The test manager will be responsible for assigning each member of the testing team to its respective areas of testing to ensure that all areas are being covered. For example, penetrating testing will be using 80 hours out of the 280 human resources hours in testing the vulnerabilities within the system, which they will report their findings (“How to Create a Test Plan”, 2018).

### *Determine Test Deliverables*

In the determine test deliverables, three testing deliverable stages must be determined before and after the product reaches its audience: before testing, during testing, and after trial. *Flow Cart 4: Before, During, After Testing Phases* describe the different testing that will be used during the system development life cycle (SDLC) (“How to Create a Test Plan”, 2018).

***Flow Cart 4: Before, During, After Testing Phases***

## **Compilation in Transit**

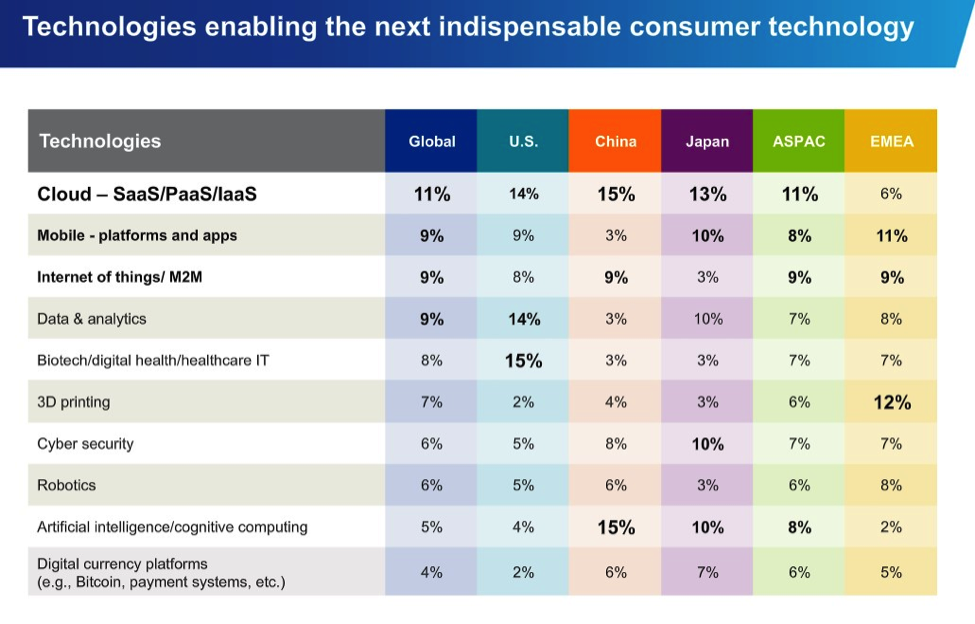
When data is in transit, there will be some compilations that may arise within the company, such as stealing information or accessing restricted areas within the network. Some of these types of security breaches come from different areas of background, and some security breaches are unintentional. Since data is on a constancy move when it is referred to as in transit, data can become very vulnerable, which is why we must develop the proper security measures to protect the data and its integrity of data.

# Deployment Strategy

## **Tailoring of Data Protections for the SaaS Cloud Infrastructure**

Cloud computing–which includes Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS)–will comprise 11% of overall technological disruption that enables “the next indispensable consumer technology,” according to KPMG’s 2015 Global Innovation Survey. SaaS dominates the cloud computing space as the most popular implementation type. Cloud computing is expected to grow at a staggering rate through 2022.

Many factors are driving the adoption of SaaS, including the benefits of rapid deployment and rapid ROI, less upfront capital investment, and a decreased reliance on limited implementation resources. Issues that can impede a successful SaaS implementation include data security, privacy, internet connectivity, and availability. The data security and privacy issues and potential solutions will be explored below. (The State of SaaS in 2015, 2016).



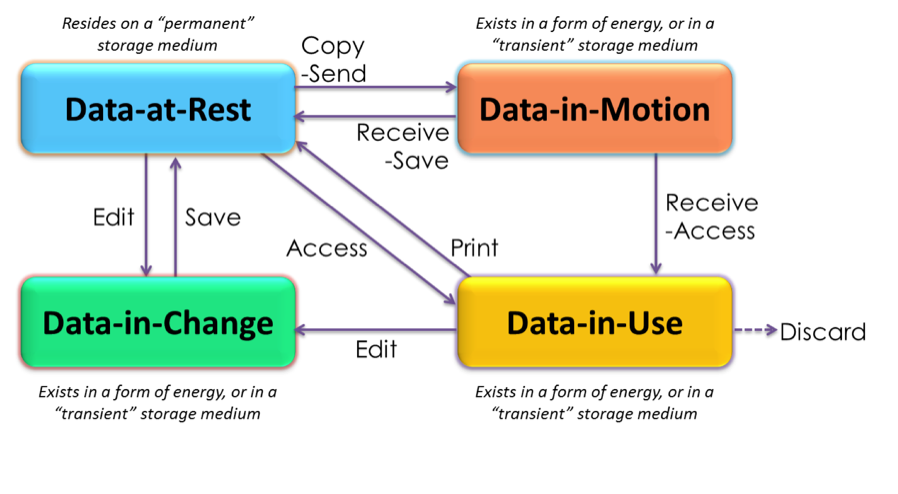
***Figure 3: Technologies enabling the next indispensable consumer technology*** (Source: KPMG‘s 2015 Global Innovation Survey)

## **Security for Cloud Computing**

For any data/information system, the best practices for securing it dictate that protection should always be in place and from every attack vector. There are four security areas of concern each with their unique requirements. These four areas when securing the cloud: the application, infrastructure, the process/storage, and personal security. This paper will focus on application and infrastructure security.

### *Access Control*

When a user utilizes a cloud SaaS service, the need for security is required immediately. Authentication is the mechanism of choice. Authentication ensures that only valid user credentials would obtain access.



***Figure 4: 4 Different stages of Data***

Data needs to be protected in three states: at rest, in use, and in motion. Each state presents unique security challenges.

### *Data at Rest*

Data is at rest when it is stored on a hard drive. Organizations need additional layers of defense to protect sensitive data from intruders if the network is compromised. Encrypting hard drives is one of the best ways to ensure the security of data at rest. Other steps can also help, such as storing individual data elements in separate locations to decrease the likelihood of attackers gaining enough information to commit fraud or other crimes.

### *Data in Use*

Data in use is more vulnerable than data at rest because, by definition, it must be accessible to those who need it. The keys to securing data in use are to control access as tightly as possible and to incorporate an authentication mechanism to ensure that users are not hiding behind stolen identities. Organizations also need to be able to track and report relevant information, so they can detect suspicious activity, diagnose potential threats, and proactively improve security. For example, an account being disabled due to a certain number of failed login attempts could be a warning sign that a system is under attack.

### *Data in Motion*

Data is at its most vulnerable when it is in motion and protecting information in this state requires specialized capabilities. Anyone with the right tools can intercept data as it moves along this path (Man in the Middle). There are several effective ways to make data in motion more secure. The best way to ensure that messages and attachments remain confidential is to transmit them through an encryption platform that integrates with existing systems and workflows. (Janacek, 2015) Other security concerns to be considered in the implementation of a SaaS cloud-based system are:

* What types of data does the organization store, use or transmit?
* Who needs to access this data?
* How is access to data controlled?
* What mechanisms are used to transport data?
* What are the pertinent laws, regulations, and standard?

Once the potential risks are identified, work with security experts and cloud vendors to ensure appropriate data security mechanisms are in place before the final implementation of a SaaS-based cloud system.

# Operations and Maintenance

One of the vital security plans that need to be documented and implemented within the network is a disaster recovery plan (DRP). A DRP is a document business plan that demonstrates the necessary actions that the company must take to resume quickly and effectively when a disaster occurs to the information technology. The main objective of this plan is to recover important data and resources that will help the company to continue its business operation until the company is fully restored (What is Disaster Recovery Plan (DRP,2013)). There are fifteen important steps that are used to provide security and maintain the business operation, which is broken down into four different categories (Mata, 2014):

* Evaluate Business Activities
* Determine Recovery Time Requirements
* Test Hypothesis
* Policies & Procedures

## **Evaluate Business Activities**

During this step, there are important activities that the company must identify to ensure the IT infrastructure is properly secured:

* Identify Critical Business Processes
* Label Dependencies
* Define Vital Applications
* Assess Current Data Recovery

### *Identify Critical Business Processes*

Superior Banking, when addressing a new security plan, such as a DRP, we need to define the data that needs to be protected and discuss the time frame that the company’s main computer system can function encase a disaster occurs to the computer system. When discussing the main computer system’s resources and how long will the company survive, we will be able to develop a more secure strategy that will help maintain its operation.

### *Label Dependencies*

The company will outline documentations about the applications that are commonly used within the system and diagnose it outcome encase it stop working. For example, if Superior Banking’s website stops working or allowing access to financial records, we must define a what-if condition that shows us the outcome and its downtime. The document should display all the applications that are being used by the company and the maximum downtime from higher downtime to lower downtime.

### *Define Vital Applications*

Once Superior Banking can identify and label all the applications that are being used to maintain its operation, as well as its maximum downtime, we will be able to define the important applications that are needed. Superior Banking will be using Software as a Service (SaaS) cloud computing infrastructure as its main application. Documenting all the vital applications that are associated with the network and SaaS cloud computing technology. SaaS will allow us to develop a plan that protects these applications.

### *Assess Current Data Recovery*

Superior Banking should already be equipped with the latest backup and recovery software that is needed to protect the data. We must understand our current data recovery strategy before we determine what is needed for improvement.

## **Determine Recovery Time Requirements**

Once we have gathered all the information that is needed, we will determine the recovery time that is required to maintain Superior Banking’s operation. Determining the amount of recovery time is an important part within the business, which address five different elements in accomplishing this task:

* Perform a Business Impact Analysis (BIA)
* Define Recovery Point Objectives (RPO)
* Distinguish Recovery Time Objectives (RTO)
* Determine Work Recovery Time (WRT)
* Designate Maximum Tolerable Downtime (MTD)

### *Perform a Business Impact Analysis (BIA)*

The business impact analysis (BIA) is the process where it will help to determine and evaluate the threats that can occur to the business operations. The BIA includes an investigation of the different computer components that reveal any vulnerabilities within the system, as well as providing strategies that help to minimize the risks. BIA is known as a business impact analysis report. The main objective of a BIA has within a disaster recovery plan is to identify costs that are linked to company’s equipment being compromised, replaces of equipment, amount of money to catch up, the profits loss, and its staff members, as well as its data. *Table 7: Elements of a BIA* address the highs and lows if a security breach or an unforeseen force had occurred within the company to stop its operation (Rouse & Sliwa, 2015).

***Table 7: Elements of a BIA***

|  |  |  |  |
| --- | --- | --- | --- |
| *Elements of a business impact analysis (BIA)* | | | |
| *Business activity affected* | ***Potential operational loss*** | ***Potential financial loss*** | ***Minimum time needed to recover operations*** |
| All services in the SaaS cloud computing infrastructure | Services is temporary unavailable | 4,000 to 6,000 revenue loss/hour | Three to four hours |
| Loss of specialized staff members due to the security breach | Activities that require them to maintain operation | Reduced ability to function normally | None, assuming backup staff is available |

### *Define Recovery Point Objectives (RPO)*

Recovery point objective (RPO) is the age of files that can be recovered and useful to maintain the business operation encase the network goes down because the result of hardware, software, human errors, and other communication are no longer available. When a computer system experience problem within the operating system, the end user will sometimes use a recovery disk to go back in time to its original state of operation. An RPO has the same qualities within the company to make sure that its business operation functions normally. RPO is an essential part of the company because it determines whether the business can stay afloat or will it sink (Rouse, 2011).

### *Distinguish Recovery Time Objectives (RTO)*

The next step will be to distinguish recover time objectives (RTO), which is another important step in the DRP. In a recovery time objective (RTO), we address the maximum tolerable and accessible length of time that the network is unresponsive. RTO helps us to define the necessary steps needed to restore the information. For example, if the RTO for the network is about two hours, we will need to gather information that will help to recover data loss in that two-hour timeframe. If the RTO for the network to be up and running will be three or more days due to holidays and the weekend, we will use a different approach that can recover information during that time loss. It’s like filling in the gaps with backup data from a time before the unforeseen accident occurred to the network (Rouse, .2011)

### *Determine Work Recovery Time (WRT)*

Work recovery time (WRT) is used to help to calculate the designate maximum tolerable downtime (MTD) that is needed to recover the important bases of the operation. Functions include checking database and logs for backup files for recover purpose, as well as making sure that all backup applications are available and running correctly (“RPO, RTO, WRT, MTD…WTH?! |Default Reasoning”, 2013).

### *Designate Maximum Tolerable Downtime (MTD)*

Designate is another word that means elect or select the maximum tolerable downtime (MTD) that the company should experience until the operation is functional. Maximum tolerable downtime (MTD) deals with the time after the incident occurs Superior Banking’s network. The expected timeframe from RTO and WRT sum will help to determine the MTD needed to ensure that data is protected and business operation can resume production.

***Flow Chart 5: RPO, RTO, WRT, & MTD Timeframe***

## **Test Hypothesis**

Gathering data and determining the recovery time needed to maintain operation is very important but testing the theory against the real threat, well that is a horse of a different color. During this part of the DRP, we will need to address the security threats and run scenario testing that relates to real-life threats to the network. Once we are able to pinpoint these threats, we will be able to develop the proper security measures against these threats. During this action, four key elements \use that help the company to determine the risk and provide the necessary solutions in restoring order within Superior Banking:

* Assess Risk
* Test Theory
* Redesign Accordingly
* Implement Solutions

### *Assess Risk*

During this step, Superior Banking will need to document the possible risk and its outcome that single points of failure that can occur to the company’s resources, such as the loss of data. *Table 8: Risk/Impact Chart of Data Loss* describes the different level of risks from low to high.

***Table 8: Risk/Impact Chart of Data Loss***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | Potential Consequences | | | | |
| L1 | L2 | L3 | L4 | L5 |
| Little or no impact to the loss of data | Minor impact to the loss of data | Moderate impact on the loss of data | Significantly impact to the loss of data | Highest impact to the loss of data |
| Not Significant | Minor | Medium | Major | Severe |
| Unlikelihood | >70% | Almost Certain | Medium | High | Very High | Very High | Very High |
| >50% | Likely | Medium | High | High | Very High | Very High |
| >25 | Possible | Low | Medium | High | High | Very High |
| >=10% | Unlikely | Low | Low | Medium | Medium | High |
| <=10% | Rare | Low | Low | Low | Low | Medium |

### *Test Theory*

The next important step is to run DRP scenarios of different incidents that can occur to the network. Performing a technology gap analysis of the current network will help us to develop the desired RPO, RTO, and MTD that is needed to ensure the operation beginning running at a distinct time.

### *Redesign Accordingly*

Once we had run a DRP scenario on the different incidents that can occur and perform a technology gap analysis of our current network, it’s time to adopted additional security if the old solutions are outdated. In computer terms, there is a saying about old technology, “Garbage In and Garbage Out”, which means that if something is old or we are using the same outdated technology with the same results, it’s time to try something new. ‘Garbage In and Garbage Out” is a major problem with companies because they don’t want to spend the necessary resources to upgrade their technology and improve their security. Making sure that the necessary investments in technology are used to close the gaps and address the different security risks to the network.

### *Implement Solutions*

The next step is developing an implementation timeline that is used to outline the plan of implementation of the new solutions. This report will describe the different technology, its purpose, the pros/cons of implementation, and the areas of security that the new solution will provide within the DRP.

## **Policies & Procedures**

It is okay to have a policy within a policy because it makes it more secured. Superior Banking needs to design a policy that helps to protect the network. One of the policies that are recommended is a network security policy (NSP). The network security policy (NSP) address the principles, procedures, and guidelines that are enforced in providing security within the network (“What is a Network Security Policy? – Definition from Techopedia”, 2018). Developing and establishing the policies and procedures is the final step in providing a successful DRP:

* Develop Procedures
* Align Procedures
* Form a Team

### *Develop Procedures*

Developing and creating an Emergency Response Policy and Procedure allows the organizing to respond to a disaster that occurs to the computer system, which helps the company to achieve full recovery of the business operation.

### *Align Procedures*

During this step, we will define severity definitions and assign escalation rules of procedures that help meet the DRP timeframe requirements.

### *Form a Team*

Forming a team is an essential part in ensuring that similar incidents don’t occur within the company. There need to be a security awareness mandatory meeting and training that is needed on a yearly basis or when the company decided to adopt new technology into its network. This type of training is the process of training and educating individuals on the proper ways of providing security awareness related to computer security (Rouse, 2011).

## **Auditing Plan for Security**

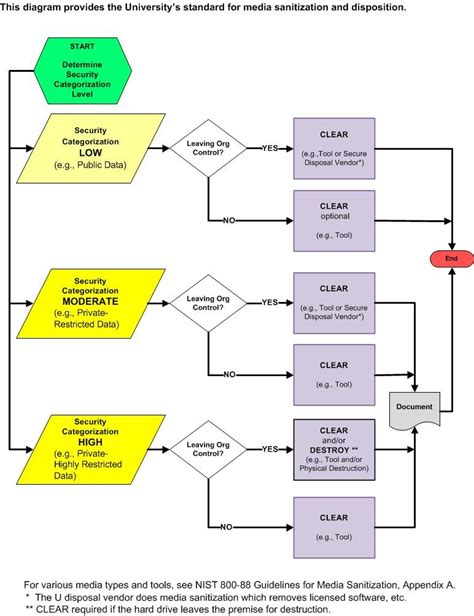
An auditing plan deals with examining the company’s resource and records to make sure that all the records are updated, as well as following different regulations that have been put into place to maintain the security of the company. The auditing plan for security is an ongoing process that needs to follow in managing the operations and maintenance of the SaaS cloud computing infrastructure. *Figure 1: Audit Continuous Plan* display a cycle on the audit planning process works, which must be established yearly to ensure that all hardware and software applications are running smoothly, as well as appropriately documented (de Aquino, 2008).

*Flow Cart 6: IT Security Audit Lifecycle*

Making sure that security within the company is well updated is paramount especially when it comes to its infrastructure.

# Disposal Plan

The final stage of the SDLC is the technology disposal process. This process is fundamental, if incorrectly implemented, can lead to regulatory non-compliance and risks of security breaches. A technology disposal program is an integral part of an overall Technology Asset Management Strategy.



***Figure 5: Security Disposal Process***

## **Technology Disposal Concerns**

### *Avoiding Data Breaches*

Every day there is a story in the news about data security, the importance of safeguarding our data and the adverse outcomes of security breaches. With the advancement of cloud computing and its increased use, this discussion is often about data security and whether data is safe in the cloud. Not only do we have to focus on where to store data, implementation security, and its use, but data needs to be secured during the disposal process as well. Data is often left on old IT assets even after it has been deleted leaving valuable data to the next user of that device.

Also, deleting files is not enough; they must be appropriately disposed. Whether data disposal is through an outsourcing partner or done in-house, it must be handled with the utmost care. Like any IT program or implementation, there are several options for disposal, and each will depend on the individual organization and its goals. Organizations who have remote employees face additional data security issues as they have employees accessing the network from outside of the office – remote data erasure does not always work – therefore, data must be wiped from every location, not just the office server room, but individual devices as well. When an outsourcing partner is destroying data, be sure to verify that the information is 100% destroyed. (McLachlan, 2017)

### *Effective Processes, Policies, and Procedures*

Without a proper effective policy and process in place, any technology disposal program is incomplete. For this plan to work, all employees should be onboard. The best way to determine which steps need to be taken to develop or improve Technology Disposal Policies is to perform a risk assessment to understand the organization’s unique data security needs. Where are the gaps in your security? Be sure to communicate efficiently with employees the proper process that must be taken should they like to retire an asset they possess.

### *Outsources Technology Disposal*

Many organizations will choose to outsource their IT technology disposal needs by selecting a partner to take care of either all portions of some of the piece. For example, a partner that will be responsible for data cleansing or even just hardware recycling. Whether you choose to do this in-house or outsource, be sure that due diligence is performed on local and national regulations for asset disposal. There are guidelines, both regulatory and environmental, that need to be followed.

Be sure that your outsourcing partner has a solid track record with governing bodies and is aware of those regulations that pertain specifically to the location and industry your organization operates in. For example, healthcare companies that must comply with HIPAA must be cognizant of these privacy restrictions when disposing of data. (McLachlan, 2017)

Here are some basic questions to ask of your vendor:

* What information security management practices and technology do they employ? Their level of investment and attention to their own security needs is often indicative of their handling of other’s information.
* What type of insurance or bonding does the vendor carry to cover data leaks?
* What asset management system does the vendor use to track and report on the disposition of equipment effectively?
* What technology does the company use to wipe or physically destroy equipment electronically? Where does this technology exist and what steps must the equipment go through before it is destroyed?
* Can the destruction of personal assets be witnessed?
* What happens to the destroyed assets after the vendor completes its work?

## **Technology Disposal Options**

An IT organization task force should be assembled to decide the best approach to dispose of IT assets that reached the end of their lifecycle. The team should consider what goals are desired when eliminating of technology. Are the goals to save money or bring money back into the organization or reduce the number of technology redundancies?

* Re-Sell – An organization can recoup capital by selling IT assets to other organizations that may have need of it. Older equipment can also be denoted for a tax-write.
* Reuse/Recycle – Some organizations can shift older systems and software to other mundane sections of their organization where the reliability of the system is not as critical.
* Complete Destruction – In some organizations, the destruction of the technology is necessary due to the sensitive nature of the data and operation of the organization. (FBI, CIA, NSA, so forth) In these instances, storage devices are degaussed and drilled through in order to destroy the ability to access any residuals of stored data.

# Conclusion

In conclusion, Superior Banking has decided to adopt a new cloud computing service, Software as a Service (SaaS), into their network to help protect data when it’s in transit over the Internet. As a team, we describe different areas within the cloud computing system and develop a report base on the SDLC. Throughout this report, we discuss the cloud computer SDLC, as well as its ends out of the SDLC process. We research the threat models that can occur when adopting the new system.

We then address the software quality requirement engineering (SQUARE) approach that helps with developing the functional design needed to ensure the transaction runs smoothly. The next area our team research was the evaluation of technology, such as the security issues that arise in the software system and development environments. We later address the software design specifications and its development plan. During the system design specification section, we discuss the different specifications that are required for data-in-transit protection across the network.

In the software development plan, we address the software need that helps the technology to evolve and develop to improve its overall purpose. The testing and integration section of our report address different methods of protecting the data and its integrity. We develop a testing plan that ensures that all security functions within the system and the network is run correctly. The deployment strategy addresses the different ways of security during the three stages of data: data at rest, data in use, and data in motion.

The operations and maintenance address the different actions that need to be taken before, during, and after the SDLC is completed. We discuss the proper DRP, as well as an IT auditing plan that is needed encase an incident occurs to the network and its resources. The final section within this report, the disposal plan, which different ways of avoiding data breaches, effective policies, outsource the disposal of technology. The report addresses the various options for technology disposal because most companies do not properly dispose of their unused technology or data and becomes victims to dumpster divers.

# References

Boer, G. (2017, April 3). What is Scrum? - Azure DevOps. Retrieved November 12, 2018, from <https://docs.microsoft.com/en-us/azure/devops/learn/agile/what-is-scrum>

De Aquino, C. E., (2008, February 01). 2008 Six Steps to an Effective Continuous Audit Process. Retrieved November 20, 2018, from <https://iaonline.theiia.org/six-steps-to-an-effective-continuous-audit-process>

Faatz, D. (2018, March 12). Best Practices for Cloud Security. Retrieved November 17, 2018, from <https://insights.sei.cmu.edu/sei_blog/2018/03/best-practices-for-cloud-security.html>

FISMApedia. (2008, March 22). Transport Layer Security. Retrieved November 18, 2018, from <http://www.fismapedia.org/index.php?title=Transport_Layer_Security>

How to Create a Test Plan. (2018). Retrieved November 12, 2018, from <https://www.guru99.com/what-everybody-ought-to-know-about-test-planing.html>

IS2200. (2013, March 03). Advantages/Disadvantages of the Spiral Model. Retrieved November 17, 2018, from <https://eternalsunshineoftheismind.wordpress.com/2013/03/03/7620/>

Janacek, B. (2015, December 1). Best Practices: Securing Data at Rest, in Use, and in Motion. Retrieved from <https://www.datamotion.com/2015/12/best-practices-securing-data-at-rest-in-use-and-in-motion/>

Kryptotel. (n.d.). Encryption Algorithms. Retrieved November 18, 2018, from <https://www.kryptotel.ae/encryption-algorithms/>

Lydford, S. (2015, June 19). Cloud Application Design Considerations. Retrieved November 21, 2018, from <https://codurance.com/2015/06/19/cloud-application-design-considerations/>

Mata, W. (2014, February 06). 15 Steps for Designing a Disaster Recovery Plan (DRP). Retrieved November 19, 2018, from <https://centretechnologies.com/15-steps-designing-successful-disaster-recovery-plan/>

McLachlan, P. (2017, August 04). Software and Hardware Disposal - Be Smart with your IT Junk. Retrieved from <http://itak.iaitam.org/software-and-hardware-disposal-be-smart-with-your-it-junk/>

Mead, N. (2006, January 30). SQUARE Process | US-CERT. Retrieved November 12, 2018, from <https://www.us-cert.gov/bsi/articles/best-practices/requirements-engineering/square-process>

Networkers. (2018, May 17). Sky-high market growth driving demand for cloud infrastructure specialists. Retrieved November 12, 2018, from <https://www.networkerstechnology.com/growth-cloud-demand-infrastructure-specialists>

NIST. (2018). Publication Search. Retrieved November 17, 2018, from <https://csrc.nist.gov/publications/sp>

O'Brien, F., & Park, A. R. (2003, April 17). Retrieved from <https://www.bus.umich.edu/KresgePublic/Journals/Gartner/research/114300/114374/114374.html>

Rackspace. (2013). Reference Architecture: Enterprise Security for the Cloud. Retrieved November 12, 2018, from <http://c744563d32d0468a7cf1-2fe04d8054667ffada6c4002813eccf0.r76.cf1.rackcdn.com/downloads/pdfs/RefArchEntSecCloud1.pdf>

Rouse, M. (2011, January). What is recovery point objective (RPO)? – Definition from WhatIs.com. Retrieved November 19, 2018, from <https://whatis.techtarget.com/definition/recovery-point-objective-RPO>

Rouse, M. (2011, August). What is recovery time objective (RTO)? – Definition from WhatIs.com. Retrieved November 19, 2018, from <https://whatis.techtarget.com/definition/recovery-time-objective-RTO>

Rouse, M. (2011, November). What is security awareness training? – Definition from WhatIs.com. Retrieved November 20, 2018, from <https://searchsecurity.techtarget.com/definition/security-awareness-training>

Rouse, M., Haughn, M., & Gibilisco, S. (2014, November). What is confidentiality, integrity, and availability (CIA triad)? – Definition from WhatIs.com*.* Retrieved November 12 2018, from <https://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA>

Rouse, M., & Sliwa, C. (2015, March). What is business impact analysis (BIA)? – Definition from WhatIs.com. Retrieved November 17, 2018, from <https://searchstorage.techtarget.com/definition/business-impact-analysis>

Rouse, M., & Burke, J. (2018, September). What is VPN (virtual private network)? – Definition from WhatIs.com.Retrieved November 12, 2018, from <https://searchnetworking.techtarget.com/definition/virtual-private-network>

RPO, RTO, WRT, MTD…WTH?! | Default Reasoning. (2013, December 10). Retrieved November 20, 2018, from <https://defaultreasoning.com/2013/12/10/rpo-rto-wrt-mtdwth/>

The State of SaaS in 2015: Global Market Trends, Growth Forecasts, Industry Reports. (2016). Retrieved from <https://financesonline.com/state-saas-2015-global-market-trends-growth-forecasts-industry-reports/>

TutorialsPoint.com. (n.d.). SDLC Waterfall Model. Retrieved November 17, 2018, from <https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm>

UC Berkeley. (2018). Data Encryption in Transit Guideline. Retrieved November 18, 2018, from <https://security.berkeley.edu/data-encryption-transit-guideline>

Verizon. (2018). Data Breach Investigation Report. Retrieved November 13, 2018, from <https://enterprise.verizon.com/resources/reports/dbir/>

Violino, B. (2011, April 25). Cloud development: 9 gotchas to know before you jump in. Retrieved November 17, 2018, from <https://www.infoworld.com/article/2683270/development-environments/cloud-development--9-gotchas-to-know-before-you-jump-in.html>

What is Network Security Policy? – Definition from Techopedia. (2018). Retrieved November 19, 2018, from <https://www.techopedia.com/definition/29916/network-security-policy>