**M S Ramaiah Institute of Technology**

(An Autonomous Institute, Affiliated to VTU)

MSR Nagar, MSRIT post, Bangalore-54

**A Mid-Sem Evaluation Report on**

**Single Pane Management of Cloud Email Security**

Submitted by

**Kush Bavishi 1MS12CS049**

Under the guidance of

|  |  |  |
| --- | --- | --- |
| **Suresh Gopathy**  Technical Leader  Cisco Systems, Inc.  Bangalore | & | **S. Rajarajeswari**  Assistant Professor  Department of Computer Science & Engineering, MSRIT |



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**M.S.RAMAIAH INSTITUTE OF TECHNOLOGY**

**(Autonomous Institute, Affiliated to VTU)**

**BANGALORE-560054**

[www.msrit.edu](http://www.msrit.edu)

**Abstract**

Cisco® Cloud Email Security provides cloud-based email protection, helping organizations reduce their onsite data centre footprint and reduce costs. Cisco Cloud Email Security monitors and filters all inbound and outbound email traffic using effective policy-based data loss prevention and encryption.

The project aims at increasing CES top-line by enabling Service Providers and Partners to host CES in their Data Centres. It integrates Configuration, Reporting and Monitoring in a Single Pane. It also aims at implementing several features to improve the User Experience, user efficiency, customer satisfaction and billing visibility.

**Contents**

1. Synopsis…………………………………………………………………………………….4

2. Project Plan…………………………………………………………………………………5

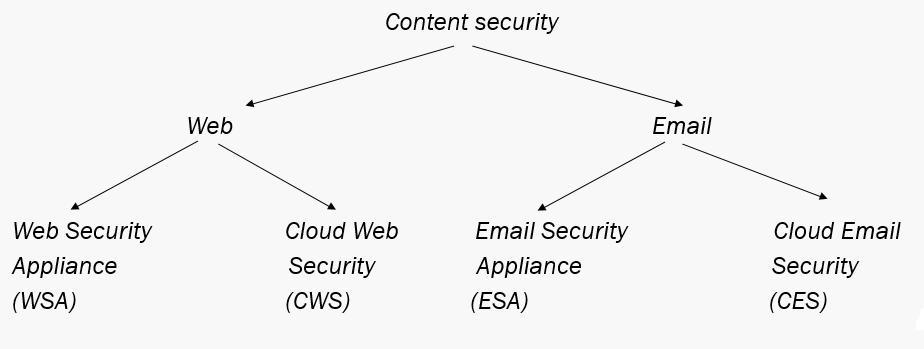
3. Software Requirement Specification………………………………………………………10

4. Literature Survey…………………………………………………………………………..13

5. Design……………………………………………………………………………………...16

6. References…………………………………………………………………………………20

**Synopsis**



**Terms and Acronyms**

* SMA – Security Management Appliance
  + - Used to connect ESAs
    - Acts as an aggregator.
* ESA – Email Security Appliance
  + - Each client has 2 ESAs in them.
    - There is a duplicate copy of each mails stored in ESAs.

Service Provider Portal provides a Single Management Pane which aims to increase Cloud Email Security (CES) topline by allowing service providers and partners to host CES in their data centres. It allows Configuration, Reporting and Monitoring of Hosted Cloud Email Service. User tracking enables more accurate billing to Service Provider and provides visibility to CISCO and its customers.

**Impact of the Developed System**

The proposed system –

* + *Acts as a single pane window (dashboard) for multiple clients.*
  + *Provides easy access to monitoring and reporting.*
  + *Configurations can be updated for individual clients.*
  + *Client specific details provided.*
  + *Easy to maintain at a single point.*

**Project Plan**

**SCRUM MODEL**

Scrum is an iterative and incremental [agile software development](https://en.wikipedia.org/wiki/Agile_software_development) framework for managing product development. It defines "a flexible, [holistic](https://en.wikipedia.org/wiki/Holism) product development strategy where a development team works as a unit to reach a common goal", challenges assumptions of the "traditional, sequential approach" to product development, and enables teams to self-organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines in the project.

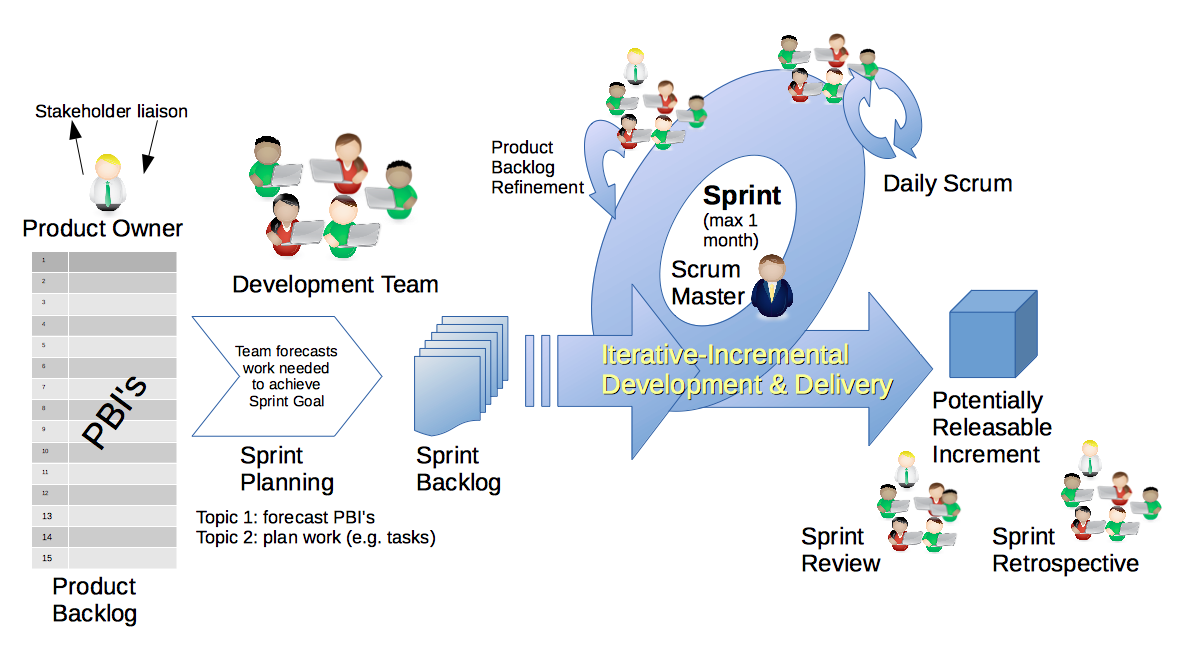
A key principle of scrum is its recognition that during production processes, the customers can change their minds about what they want and need (often called requirements volatility), and that unpredicted challenges cannot be easily addressed in a traditional predictive or planned manner. As such, scrum adopts an [empirical](https://en.wikipedia.org/wiki/Empirical) approach—accepting that the problem cannot be fully understood or defined, focusing instead on maximizing the team's ability to deliver quickly, to respond to emerging requirements and to adapt to evolving technologies and changes in market conditions.

**Workflow**

A sprint (or iteration) is the basic unit of development in scrum. The sprint is a [*time boxed*](https://en.wikipedia.org/wiki/Timeboxing) effort; that is, it is restricted to a specific duration. The duration is fixed in advance for each sprint and is normally between one week and one month, with two weeks being the most common.

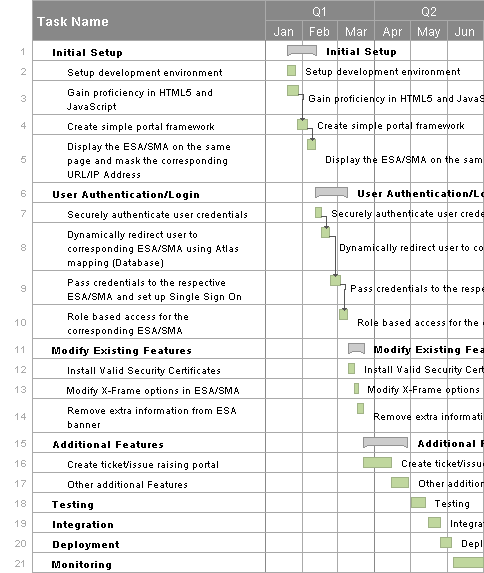
Each sprint starts with a sprint planning event that aims to define a sprint backlog, identify the work for the sprint, and make an estimated commitment for the sprint goal. Each sprint ends with a sprint review and sprint retrospective, that reviews progress to show to stakeholders and identify lessons and improvements for the next sprints.

Scrum emphasizes working product at the end of the sprint that is really *done*. In the case of software, this likely includes that the software has been integrated, fully tested, end-user documented, and is potentially shippable.



**Fig. Scrum Workflow**

**PROJECT SCHEDULE USING GANTT CHART**

****

**TOP 5 RISKS AND MITIGATION STEPS**

## 1. Inherent Schedule Flaws

Software development, given the intangible nature and uniqueness of software, is inherently difficult to estimate and schedule.

**Mitigation**: On agile projects the team is heavily involved in planning and estimating through activities such as XP's planning game and Wideband Delphi workshops. By working in short increments the true velocity of the team quickly emerges and is visible to all stakeholders who are now more closely involved in the project.

**2. Budget Risk**

A project poses budget risk if there is:

* Wrong estimation of cost
* Cost overrun
* Project scope expansion

**Mitigation:** To undertake better effort estimation. Building a COCOMO model can mitigate the budget risk. Build an SRS model.

## 3. Requirements Inflation

As the project progresses more and more features that were not identified at the beginning of the project emerge that threaten estimates and timelines.

**Mitigation:** Agile projects plan in the regular trade-off discussions about features and estimates at every iteration boundary. Changes and requirements inflation are accepted as a fact of software projects. Rather than utilizing change-suppression mechanisms, prioritization sessions are scheduled that allow worthwhile changes to proceed and initially envisioned features to be superseded if the business gives their authorization. It has never been possible to squeeze a pint into a quart cup, but now at least we anticipate the likely issue and have mechanisms in place to address the matter as part of the project from its early stages.

**4. Operational Risk**

Risks of loss due to improper process implementation, failed system or some external events risks.  
Causes of Operational risks:

* Failure to address priority conflicts
* Failure to resolve the responsibilities
* Insufficient resources
* No proper subject training
* No resource planning
* No communication in team.

**Mitigation**: Assign project to the person having skills required for it, Continuous training of skills, Communication between teams.

**5. Technical risks:**

Technical risks generally leads to failure of functionality and performance.  
Causes of technical risks are:

* Continuous changing requirements
* No advanced technology available or the existing technology is in initial stages.
* Product is complex to implement.
* Difficult project modules integration.

**Mitigation**: Using of SDLC such as Agile which can overcome the issue of continuous changing requirements, breaking the product to make it less complex.

**Software Requirement Specification**

1. **Project Overview**

* Objective: To build a single pane portal for both clients and internal use.
* The clients can be hosted on any cloud storage such as Azure or AWS.
* The single pane access provides the following functions:
  1. Configuration
  2. Reporting
  3. Monitoring
* The proposed system:

1. Acts as a single pane window (dashboard) for multiple clients.
2. Provides easy access to monitoring and reporting.
3. Configurations can be updated for individual clients.
4. Determines and displays the respective ESA/SMA units for each client.
5. Contains all three units i.e. configuration, monitoring & reporting unit in it which features data loss prevention technology & powerful encryption technology delivering powerful business-class email features.
6. Masks the URL/IP of the corresponding configuration, monitoring and reporting units, therefore providing higher security.
7. **External Interface Requirements**

**2.1 User Interfaces**

The users will interact with a web UI through their web browsers.

**2.2 Hardware Interfaces**

* + Client Side

Any computer with capabilities to run an operating system and a modern browser. Recommended – RAM: 512GB, Processor: Intel/AMD 1.2 GHz.

* + Server Side

Server with high uptime and high resources to be able to process several requests simultaneously.

**2.3 Software Interfaces**

* + Client Side
    - Web Browser (any)
    - Operating System (any)
  + Server Side
    - MySQL Database
    - Operating System (any)
    - Web Browser
    - IDE’s for developing the web pages (HTML, JavaScript, PHP)

**2.4 Communication Interfaces**

* + Client will access the portal using HTTP/HTTPS protocol.

1. **Functional Requirements**
   1. Provide Single Sign On feature so that the client needs to log in once, on the portal, and gets automatically logged in to the respective ESAs and SMAs.
   2. Retrieve and display only the units which belong to the respective clients.
   3. Provide all of the clients units (configuring, reporting, monitoring) at the same place, accessible by just a single click.
   4. Allow clients to use their own user databases and allow external authentication.
   5. Internally redirect clients to their respective units, therefore masking the URLs of the ESA/SMA units and ensuring higher levels of security.
   6. Count the number of unique email users under a particular domain, for Smart Licencing.
   7. Count the number of inbound and outbound emails per unique user and also report the highest email senders/recipients.
2. **Software System Attributes**

*4.1 Reliability*

The application will meet all of the functional requirements without any unexpected behaviour.

*4.2 Availability*

The application will be hosted as a cloud service and will be available to the client on demand and at all times.

*4.3 Security*

The application adds a layer of security on the existing Security applications, thus increasing the security significantly.

*4.4 Portability*

The portal will run on any operating system with a browser and an active internet connection.

*4.5 Maintainability*

The application will be written clearly and concisely. The code will be well documented. Particular care will be taken to design the project modularly to ensure that maintenance is easy.

1. **Performance Requirements**

5.1 Real time performance:

The system should be able to look up the users and map the ESA/SMA in real time.

5.2 Parallelism:

The application must be able to process several requests simultaneously.

5.3 Availability:

The server should have 99.99% up time and be available to the clients on demand.

1. **Database Requirements**

The application will use a MySQL database in the backend to validate the clients and also to map the corresponding ESA/SMA units to the given client. Also, the application will

1. **Design Constraints**

The design constraints that will affect the design of this application are the schedule and the performance. The project must be completed on schedule and the performance of the application must be very high.

**Literature Survey**

**INTRODUCTION**

Cisco® Cloud Email Security provides cloud-based email protection, helping organizations reduce their onsite data centre footprint and reduce costs.

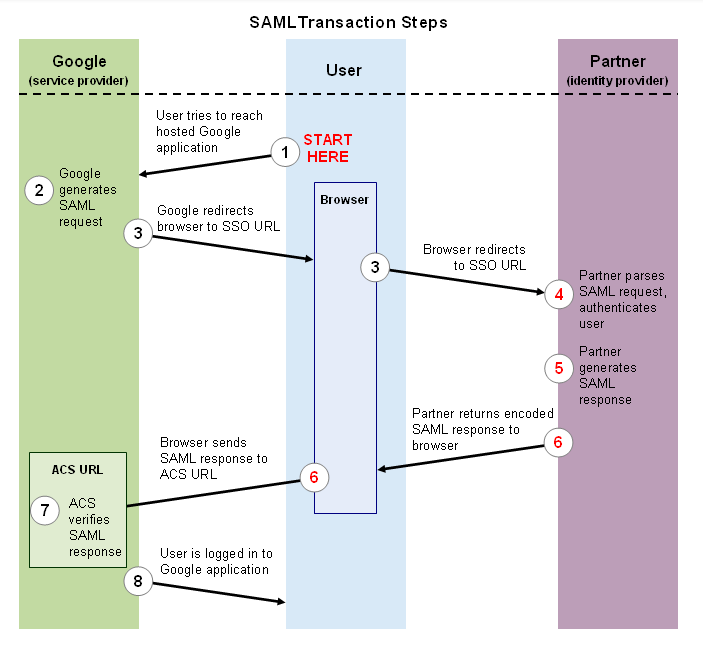
Cisco Cloud Email Security monitors and filters all inbound and outbound email traffic using effective policy-based data loss prevention and encryption. Its sophisticated filter strategy blocks targeted attacks using reputation, malware, and outbreak filters. Cisco Cloud Email Security uses the power of Cisco Security Talos, the largest threat detection network in the world, to provide proven, zeroday threat protection to all users wherever they are. It monitors and detects threats from:

* Over 75 TB of web data per day
* More than 1.6 million deployed devices
* More than 150 million endpoints
* More than 13 billion web requests per day
* Over 35 percent of the world’s email traffic

**METHOD**

The project tries to integrate all the Cloud Email Security Devices of a particular user into one single portal. It aims at increasing CES top-line by enabling Service Providers and Partners to host CES in their Data Centres. Currently, all devices are available on different URL’s/Domains. After the creation of this portal, Configuration (ESA – Email Security Appliance), Reporting (SMA – Security Management Appliance) and Monitoring, will all be available from a single portal. Also, security is being increased by masking the URLs of the customers’ devices, therefore reducing the chances of possible hacking attacks.

Besides integrating the devices, the portal also aims at implementing Single Sign-On using SAML. Single sign-on (SSO) is a [session](http://searchsoa.techtarget.com/definition/session)/user [authentication](http://searchsecurity.techtarget.com/definition/authentication) process that permits a user to enter one name and password in order to access multiple applications. The process authenticates the user for all the applications they have been given rights to and eliminates further prompts when they switch applications during a particular session.This greatly reduces the amount of credentials that the admins need to remember and also increases the overall efficiency. The concept on Single Sign-On is being used extensively today to improve user experience. Major companies such as Google as implanting it. The most common example is the services/apps provided by Google on the web. A single log in to the Google page provides access to all of Google’s websites, such as Gmail, YouTube, etc, without having to log in again on any of those pages. It is proven to increase efficiency and provide better User Experience for seamless access to all services.



The project also aims at implementing Smart Licencing. This greatly enhances the ability of the Cisco Cloud Email Security as it provides a scalable service to customers. Currently, there is no system to be able to track the actual usage and unique users of a system. This feature tracks the unique users/usage and provide visibility to Service Providers as well as Cisco and its customers. It helps billing the clients based on their usage. The client need not pay a fixed amount for the fixed service. This increases flexibility and also boosts business for Cisco and its customers.

**CONCLUSION**

The study highlights the details of Single Pane Management of Cloud Email Security and brings out the benefits of integrating Configuration, Monitoring and Reporting, facilitating single sign on and providing user tracking for Smart Licensing and better billing of customers. Thus there will be higher cost efficiency on the customer side and also the user experience and efficiency for using Cloud Email Security.

**REFERENCES**

* https://developers.google.com/google-apps/sso/saml\_reference\_implementation
* http://www.cisco.com/c/en/us/td/docs/voice\_ip\_comm/cucm/admin/8\_5\_1/ccmfeat/fsgd-851-cm/fssso.html
* http://mkchendil.blogspot.in/2015/03/okta-integration-with-apache-usingmodauthmellon.html
* https://wiki.python.org/moin/PythonTraining
* https://wiki.cisco.com

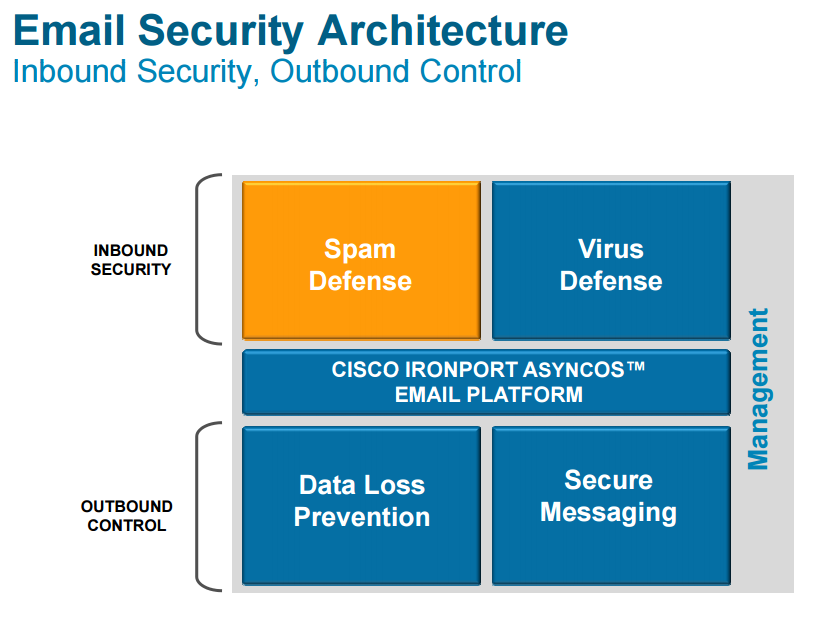
**Design**

**INTRODUCTION**

**Organization of the Document**

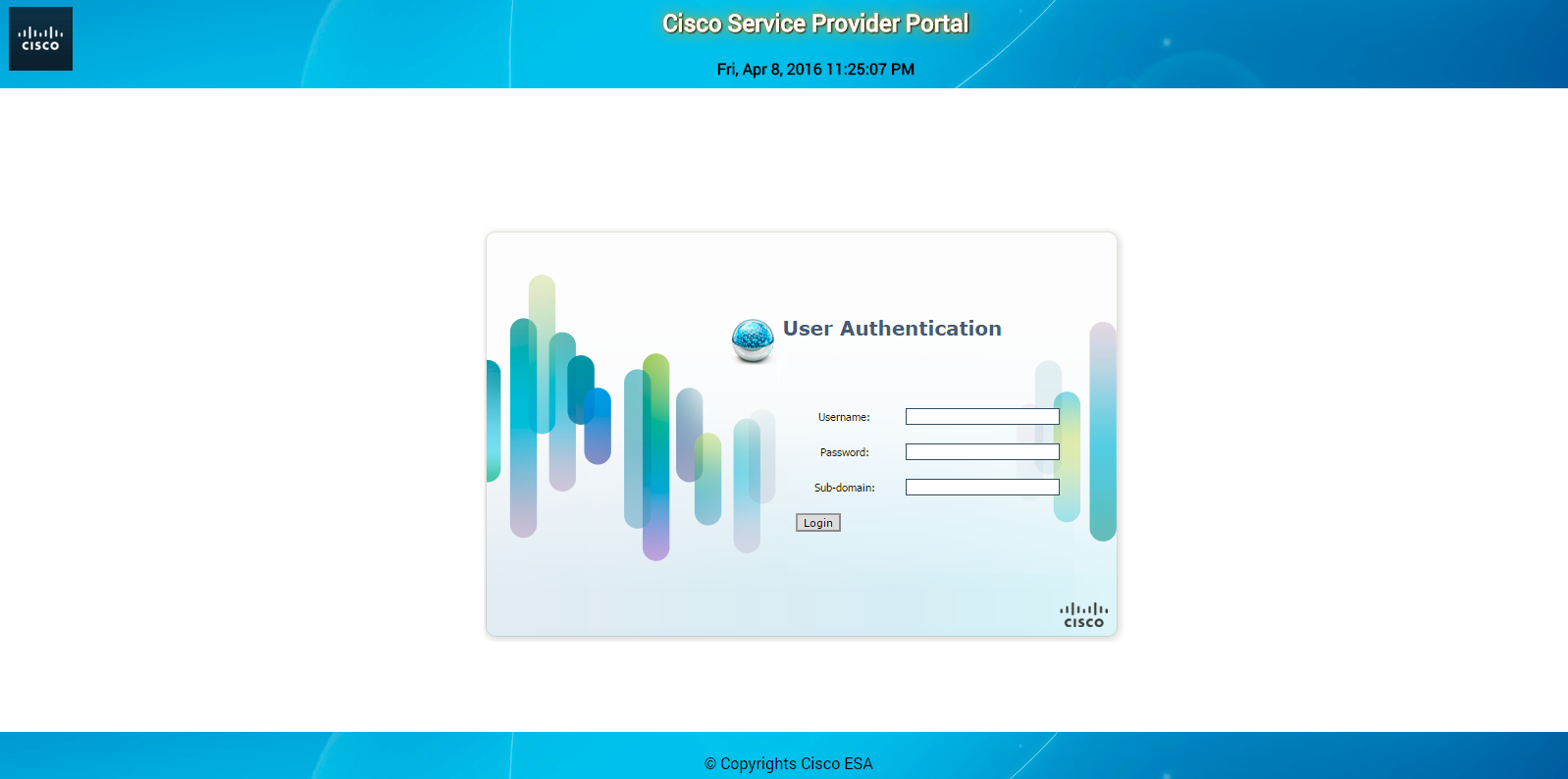
1. Architecture Design
2. Graphical User Interface
3. Sequence Diagram
4. References

**ARCHITECTURE DESIGN**

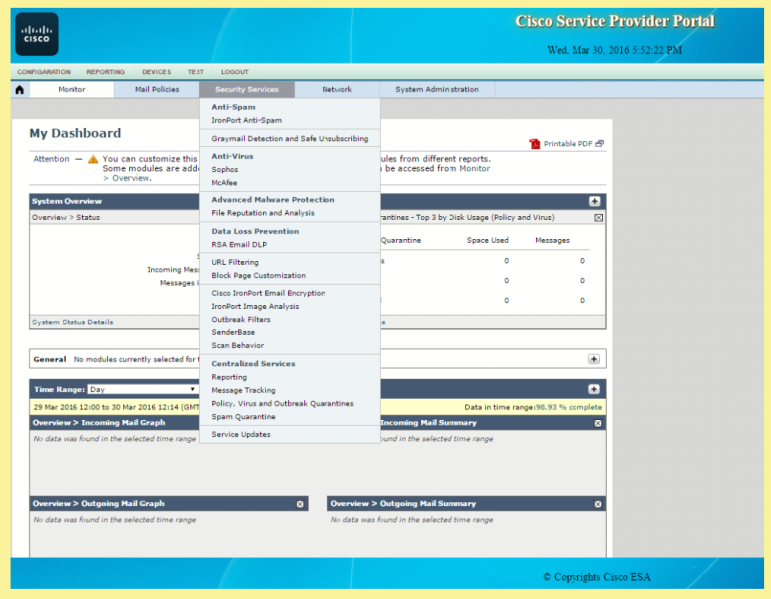


**Fig. Email Security Architecture**

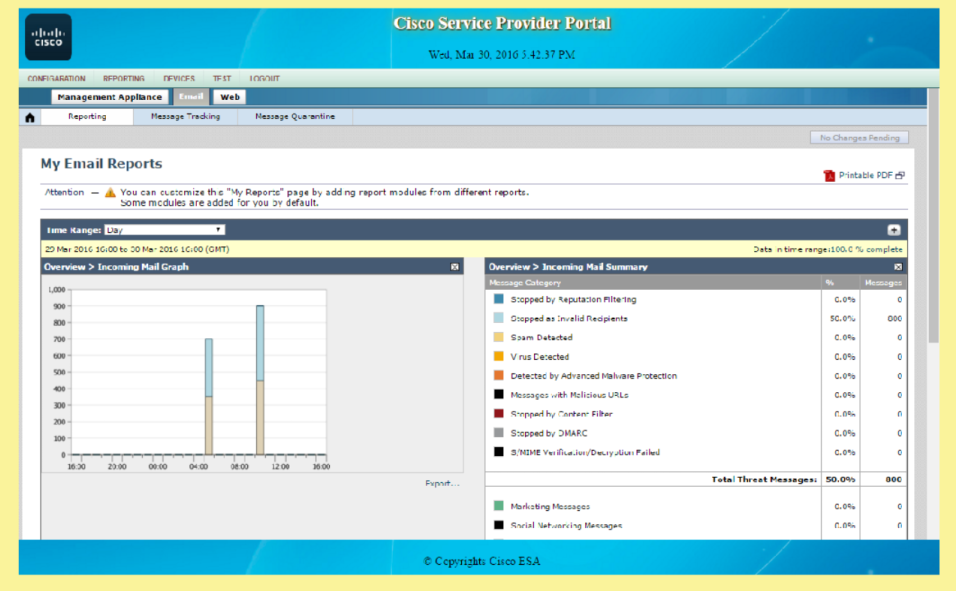
**GRAPHICAL USER INTERFACE**



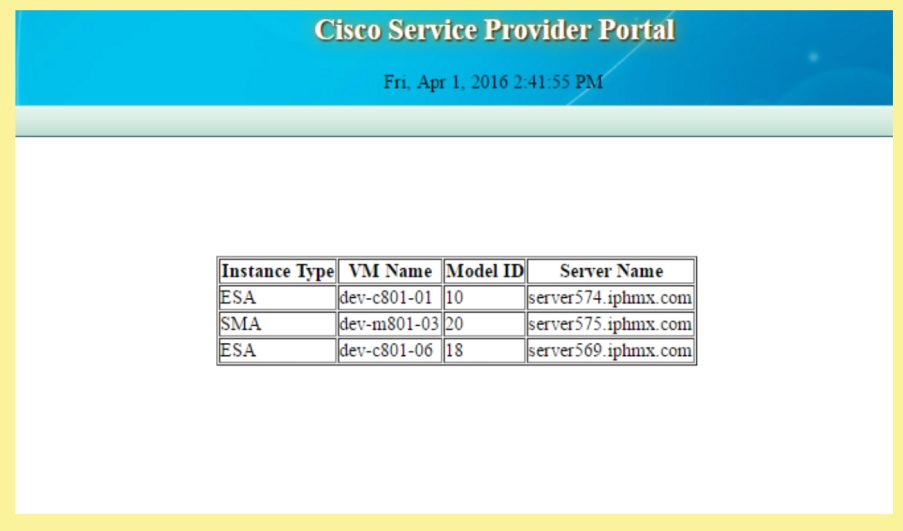
**Fig. Login Page**



**Fig. Configuration Page**

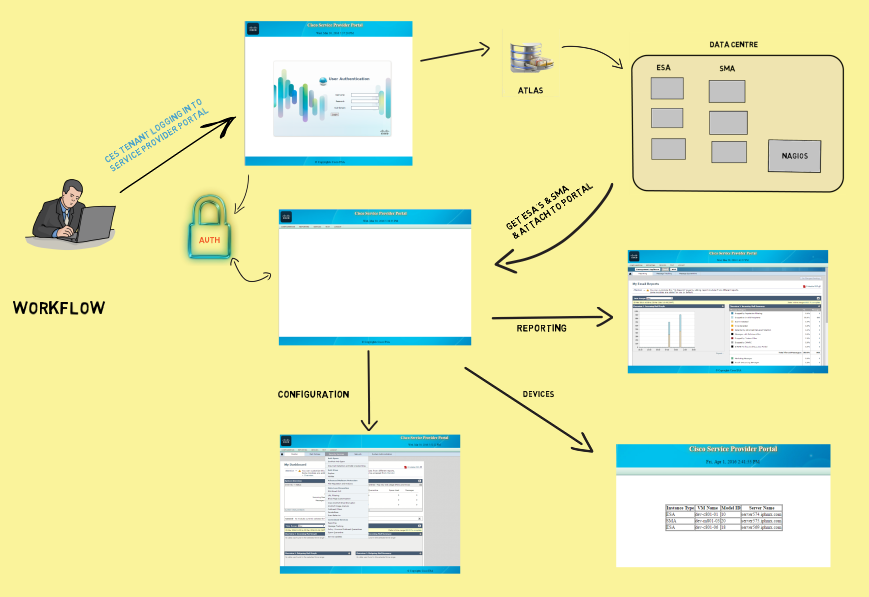


**Fig. Reporting Page**

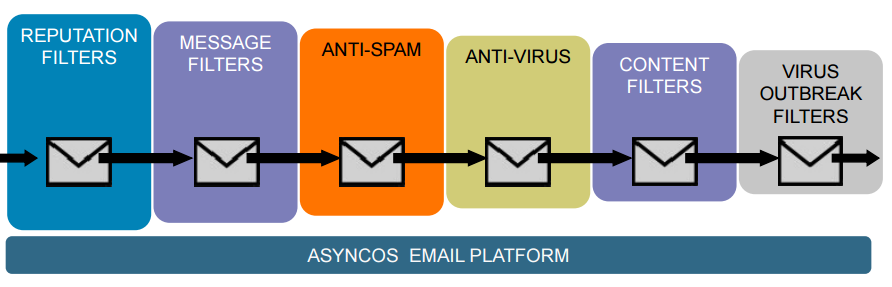


**Fig. Monitoring Page**

**SEQUENCE DIAGRAM**



**Fig. Workflow**



**Fig. Stages of Email Filtering**

**REFERENCES**

* https://www.cisco.com
* www.wikipedia.org
* https://developers.google.com/google-apps/sso/saml\_reference\_implementation
* http://www.cisco.com/c/en/us/td/docs/voice\_ip\_comm/cucm/admin/8\_5\_1/ccmfeat/fsgd-851-cm/fssso.html
* http://mkchendil.blogspot.in/2015/03/okta-integration-with-apache-usingmodauthmellon.html
* https://wiki.python.org/moin/PythonTraining
* https://wiki.cisco.com