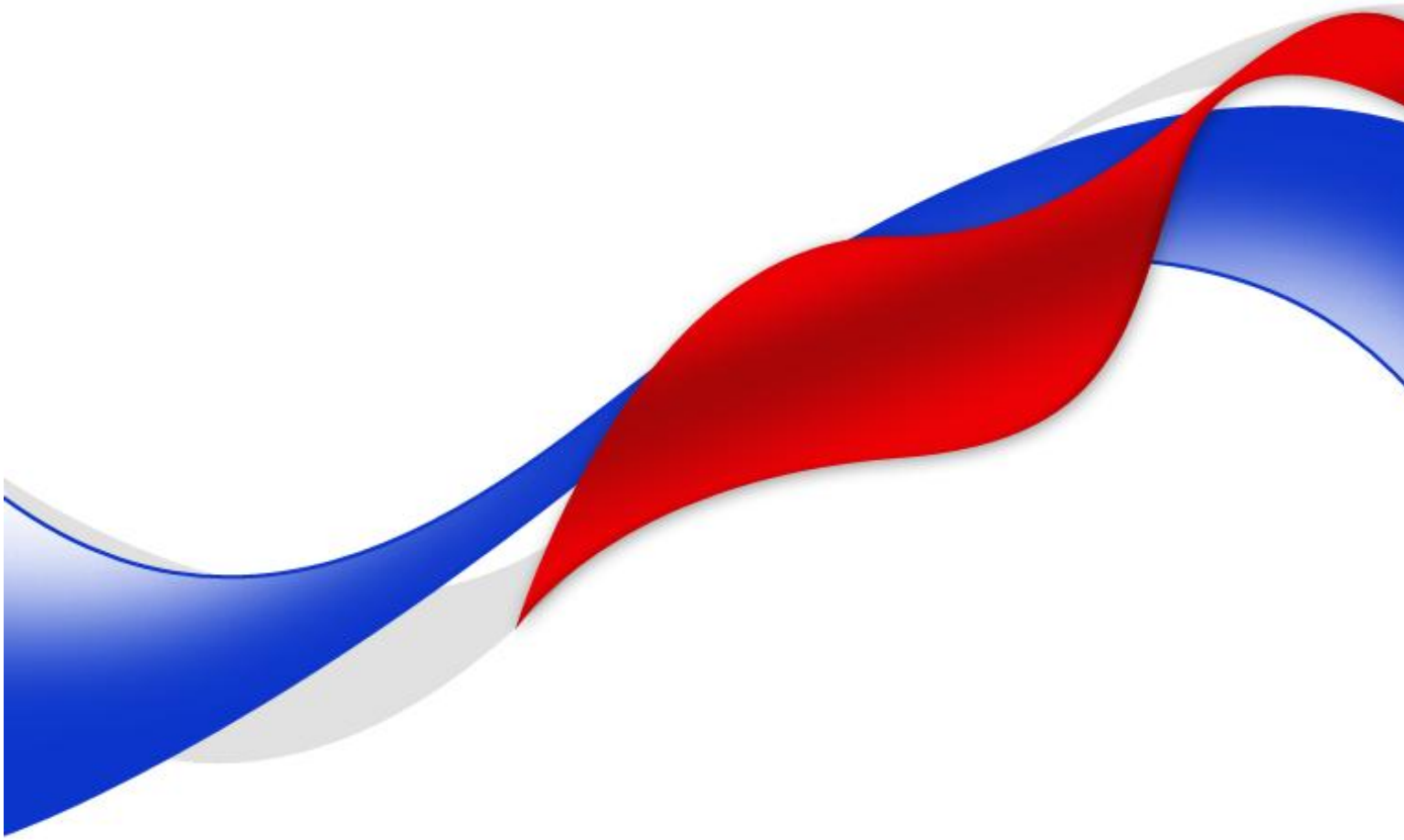
Leo TechnoSoft

Value Addition to your technology



Securing SaaS Applications: A Cloud Security Perspective for Application Providers

Software as a Service [SaaS] is rapidly emerging as the dominant delivery model for meeting the needs of enterprise IT services. However, most enterprises are still uncomfortable with the SaaS model due to lack of visibility about the way their data is stored and secured. According to the Forrester study, "The State of Enterprise Software: 2009," security concerns are the most commonly cited reason why enterprises aren't interested in SaaS. Consequently, addressing enterprise security concerns has emerged as the biggest challenge for the adoption of SaaS applications.

This article focuses on security considerations, while architecting SaaS applications, and mitigation strategies for meeting the security challenges. The adoption of these security practices can help SaaS providers instill enterprises with a degree of confidence in their security by eliminating security vulnerabilities and ensuring the safety of sensitive data.

Overview of SaaS

Software as a Service (SaaS) is a software deployment model where applications are remotely hosted by the application or service provider and made available to customers on demand, over the Internet. Enterprises can take advantage of the SaaS model to reduce the IT costs associated with traditional on-premise applications like hardware, patch management, upgrades, etc. On demand licensing can help customers adopt the "pay-as-you-go/grow" model to reduce their up-front expenses for IT purchases.

SaaS lets software vendors control and limit use, prohibits copies and distribution, and facilitates the control of all derivative versions of their software. SaaS centralized control often allows the vendor to establish an ongoing revenue stream with multiple businesses [tenants] and users. The tenants are provided a protected sandbox view of the application that is isolated from other tenants. Each tenant can tune the metadata of the application to provide a customized look and feel for its users.

The SaaS software vendor may host the application on its own private server farm or deploy it on a cloud computing infrastructure service provided by a third party provider (e.g. Amazon, Google, etc.). The use of cloud computing coupled with the pay-as-you-go (grow) approach helps the application service provider reduce the investment in infrastructure services and enables it to concentrate on providing better services to customers.

Security Challenges for SaaS

Over the past decade, computers have become widespread within enterprises, while IT services and computing has become a commodity. Enterprises today view data and business processes (transactions, records, pricing information, etc.) themselves as strategic and guard them with access control and compliance policies.

However, in the SaaS model, enterprise data is stored at the SaaS provider's data center, along with the data of other enterprises. Moreover, if the SaaS provider is leveraging a public cloud computing service, the enterprise data might be stored along with the data of other unrelated SaaS applications. The cloud provider might, additionally, replicate the data at multiple locations across countries for the purposes of maintaining high availability.

Most enterprises are familiar with the traditional on-premise model, where the data continues to reside within the enterprise boundary, subject to their policies. Consequently, there is a great deal of discomfort with the lack of control and knowledge of how their data is stored and secured in the SaaS model. There are strong concerns about data breaches, application vulnerabilities and availability that can lead to financial and legal liabilities.

The following figure illustrates the layered stack for a typical SaaS vendor and highlights critical aspects that must be covered across layers in order to ensure security of the enterprise data.

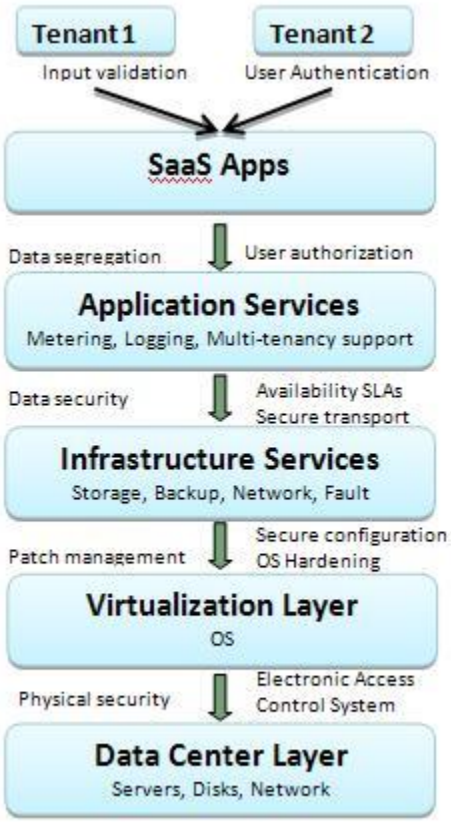


Figure 1. Security for the SaaS stack

High Level Security Considerations

The following key security elements should be carefully considered as an integral part of the SaaS application development and deployment process:

- SaaS deployment model
- Data security
- Network security
- Regulatory compliance
- Data segregation
- Availability
- Backup
- Identity management and sign-on process

SaaS Deployment Model

The SaaS security challenges differ depending upon the deployment model being used by the vendor. SaaS vendors may choose to deploy the solution either by using a public cloud vendor or host it themselves. Dedicated public cloud providers such as Amazon help to build secure SaaS solutions by providing infrastructure services that aid in ensuring perimeter and environment security. This involves the use of firewalls, intrusion detection systems, etc. A self-hosted SaaS deployment, however, requires the vendor to build these services and assess them for security vulnerabilities.

Data Security

In a traditional on-premise application deployment model, the sensitive data of each enterprise continues to reside within the enterprise boundary and is subject to its physical, logical and personnel security and access control policies. However, in the SaaS model, the enterprise data is stored outside the enterprise boundary, at the SaaS vendor end. Consequently, the SaaS vendor must adopt additional security checks to ensure data security and prevent breaches due to security vulnerabilities in the application or through malicious employees. This involves the use of strong encryption techniques for data security and fine-grained authorization to control access to data.

In cloud vendors such as Amazon, the Elastic Compute Cloud [EC2] administrators do not have access to customer instances and cannot log into the Guest OS. EC2 Administrators with a business need are required to use their individual cryptographically strong Secure Shell [SSH] keys to gain access to a host. All such accesses are logged and routinely audited. While the data at rest in Simple Storage Service [S3] is not encrypted by default, users can encrypt their data before it is uploaded to Amazon S3, so that it is not accessed or tampered with by any unauthorized party.

Network Security

In a SaaS deployment model, sensitive data is obtained from the enterprises, processed by the SaaS application and stored at the SaaS vendor end. All data flow over the network needs to be secured in order to prevent leakage of sensitive information. This involves the use of strong network traffic encryption techniques such as Secure Socket Layer [SSL] and the Transport Layer Security [TLS] for security.

In case of Amazon WebServices [AWS], the network layer provides significant protection against traditional network security issues, such as MITM attacks, IP spoofing, port scanning, packet sniffing, etc. For maximum security, Amazon S3 is accessible via SSL encrypted endpoints. The encrypted endpoints are accessible from both the Internet and from within Amazon EC2, ensuring that data is transferred securely both within AWS and to and from sources outside of AWS.

Regulatory Compliance

The SaaS deployment needs to be periodically assessed for conformance to regulatory and industry standards. The SAS 70 standard includes operating procedures for physical and perimeter security of data centers and service providers. Access, storage, and processing of sensitive data needs to be carefully controlled and is governed under regulations such as ISO-27001, Sarbanes-Oxley Act [SOX], Gramm-Leach-Bliley Act [GLBA], Health Insurance Portability and Accountability Act [HIPAA] and industry standards like Payment Card Industry Data Security Standard [PCI-DSS].

Data privacy has emerged as another significant challenge. Different countries have their distinct privacy regulations about how data needs to be secured and stored. These might lead to conflicts when the enterprise data of one country is stored in data centers located in another country.

Data Segregation

In a mature multi-tenant SaaS architecture, the application instances and data stores may be shared across multiple enterprises. This allows the SaaS vendor to make more efficient use of resources and helps achieve lower costs. At the same time, sufficient security checks need to be adopted to ensure data security and prevent unauthorized access to data of one tenant by users from other tenants. This involves hardening the data store as well as the application to ensure data segregation.

In case the SaaS application is deployed at a third party cloud vendor, additional safeguards need to be adopted so that data of an application tenant is not accessible to other applications.

In the case of Amazon, the S3 APIs provide both bucket-level and object-level access controls, with defaults that only permit authenticated access by the bucket and/or object creator. Write and Delete permission is controlled by an Access Control List (ACL) associated with the bucket. Permission to modify the bucket's ACL is itself controlled by an ACL, and it defaults to creator-only access. Therefore, the customer maintains full control over who has access to their data. Amazon S3 access can be granted based on AWS Account ID, DevPay Product ID, or open to everyone.

Availability

The SaaS application needs to ensure that enterprises are provided with service around the clock. This involves making architectural changes at the application and infrastructural levels to add scalability and high availability. A multi-tier architecture needs to be adopted, supported by a load-balanced farm of application instances, running on a variable number of servers. Resiliency to hardware/software failures, as well as to denial of service attacks, needs to be built from the ground up within the application.

At the same time, an appropriate action plan for business continuity [BC] and disaster recovery [DR] needs to be considered for any unplanned emergencies. This is essential to ensure the safety of the enterprise data and minimal downtime for enterprises.

With Amazon for instance, the AWS API endpoints are hosted on the same Internet-scale, world-class infrastructure that supports the Amazon.com retail site. Standard Distributed Denial of Service [DDoS] mitigation techniques such as syn cookies and connection limiting are used. To further mitigate the effect of potential DDoS attacks, Amazon maintains internal bandwidth that exceeds its provider-supplied Internet bandwidth.

Backup

The SaaS vendor needs to ensure that all sensitive enterprise data is regularly backed up to facilitate quick recovery in case of disasters. Also the use of strong encryption schemes to protect the backup data is recommended to prevent accidental leakage of sensitive information.

In the case of cloud vendors such as Amazon, the data at rest in S3 is not encrypted by default. The users need to separately encrypt their data and backups so that it cannot be accessed or tampered with by unauthorized parties.

Identity Management [IdM] and Sign-on Process

The SaaS vendor can support identity management and sign on services using any of the following models.

1. Independent IdM stack
   The SaaS vendor provides the complete stack of identity management and sign on services. All information related to user accounts, passwords, etc. is completely maintained at the SaaS vendor end.
2. Credential Synchronization
   The SaaS vendor supports replication of user account information and credentials between enterprise and SaaS application. The user account information creation is done separately by each tenant within the enterprise boundary to comply with its regulatory needs. Relevant portions of user account information are replicated to the SaaS vendor to provide sign on and access control capabilities. The authentication happens at the SaaS vendor end using the replicated credentials.
3. Federated IdM
   The entire user account information including credentials is managed and stored independently by each tenant. The user authentication occurs within the enterprise boundary. The identity of the user as well as certain user attributes are propagated on-demand to the SaaS vendor using federation to allow sign on and access control.

The following table highlights the security challenges for adopting these models and the relative advantages and disadvantages.

IdM and SSO Model	Advantages	Disadvantages	Security Challenges
Independent IdM stack	> Easy to implement > No separate integration with enterprise directory	> The users need to remember separate credentials for each SaaS application	> The IdM stack should be highly configurable to facilitate compliance with enterprise policies; e.g., password strength, etc.
Credential Synchronization	>Users don't need to remember multiple passwords	> Requires integration with enterprise directory > Has higher security risk value due to transmissions of user credentials outside enterprise perimeter	> The SaaS vendor needs to ensure security of the credentials during transit and storage and prevent their leakage
Federated IdM	> Users don't need to remember multiple passwords > No separate integration with enterprise directory > Low security risk value as compared to credential synch	> Relatively more complex to implement	> The SaaS vendor and tenants need to ensure that proper trust relationships and validations are established to ensure secure federation of user identities

Securing SaaS Applications

We have identified the following key mitigation strategies for addressing the above critical security challenges and improving the robustness of the SaaS applications

- Secure Product Engineering
- Secure Deployment
- Governance and Regulatory Compliance Audits
- Third-Party SaaS Security Assessment

Secure Product Engineering

Product vendors are always rushing to meet market release deadlines. Consequently, product security is often given lesser precedence. This can result in buggy software that is prone to security vulnerabilities. It is a known fact that leakage of sensitive data due to security exploits can result in heavy financial loss to enterprises and expose the SaaS vendor to potential liability issues along with lost credibility.

It is highly recommended that software vendors treat security as part of the product engineering lifecycle. At each phase of development [architecture, design, coding], a security review should be performed. This will help with faster identification of any security issues and lower rework costs for any security fixes that need to be implemented. The coding and testing guidelines should similarly be revised while keeping security considerations in perspective.

Secure Deployment

As discussed, SaaS solutions can either be hosted by the SaaS vendor or they can be deployed on a public cloud. In a self-hosted deployment, the SaaS vendor needs to ensure that adequate safeguards are adopted to combat against network penetration and DoS attacks. Dedicated cloud providers such as Amazon and Google help facilitate building secure SaaS applications by providing infrastructure services that aid in ensuring data security, network security, data segregation, etc. The SaaS applications that are deployed on these public clouds should ensure that they harden their application security settings to conform to the best practices recommended by the public cloud vendor.

Governance and Regulatory Compliance Audits

Third party Governance and Regulatory Compliance [GRC] audits can help validate the conformance of the SaaS vendors to government regulations and industry standards such as ISO27001, SOX, GLBA, HIPAA and PCI-DSS. Additionally, they can validate that appropriate BC and DR plans are in place and followed meticulously.

GRC audits help the SaaS vendor to identify and fix any deviations from regulations to ensure compliance to industry standards. They also help the SaaS provider ease customer concerns about the security, privacy and availability of the enterprise data, and help build credibility. It is recommended that SaaS vendors periodically conduct a third-party GRC audit to ensure compliance.

Third-Party SaaS Security Assessment

Third-party SaaS security assessments help validate the security and integrity of the SaaS application and its deployment. It is recommended that SaaS vendors periodically conduct a SaaS security assessment to ensure the security of their solutions.

The standard tools and techniques used for web application vulnerability assessments (VA) as captured by Open Web Application Security Project [OWASP] do not provide sufficient coverage for SaaS-specific concepts such as multi-tenancy, data segregation, etc. The Cloud Security Alliance [CSA] captures the critical areas for SaaS applications in their CSA Security Guide. A security assessment specifically tailored for SaaS solutions that incorporates these critical areas is essential for detecting security vulnerabilities and fixing them before they can be exploited by malicious hackers.

The SaaS security assessment should be comprised of both the application VA as well as network VA for complete coverage. The following figure gives an overview of the security threats and vulnerabilities which should be covered as part of the security assessment.

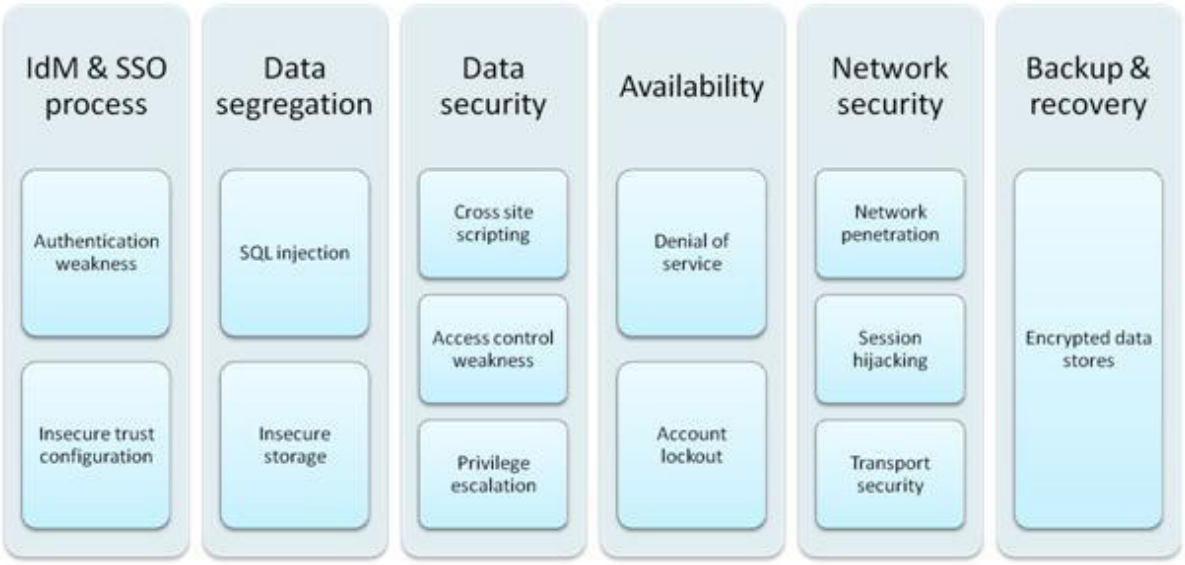


Figure 2. Security considerations and vulnerabilities

Application Vulnerability Assessment

The application VA helps validate application security in a SaaS deployment. This is generally independent of the SaaS deployment model used by the vendor. However, dedicated cloud providers such as Amazon help facilitate building secure SaaS applications by providing infrastructure services that aid in ensuring data security, network security, data segregation, etc.

Data Security
Malicious users can exploit weaknesses in the data security model to gain unauthorized access to data. The following assessments test and validate the security of the enterprise data stored at the SaaS vendor.

- Cross site scripting [XSS]
- Access control weaknesses
- OS and SQL Injection Flaws
- Cross site request forgery [CSRF]
- Cookie manipulation
- Hidden field manipulation
- Insecure storage
- Insecure configuration

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data and lead to a financial loss.

Network Security
Malicious users can exploit weaknesses in network security configuration to sniff network packets. The following assessments test and validate the network security of the SaaS vendor.

- Network penetration and packet analysis
- Session management weaknesses
- Insecure SSL trust configuration

Any vulnerability detected during these tests can be exploited to hijack active sessions, gain access to user credentials and sensitive data.

Data Segregation
A malicious user can use application vulnerabilities to handcraft parameters that bypass security checks and access sensitive data of other tenants. The following assessments test and validate the data segregation of the SaaS vendor in a multi-tenant deployment.

- SQL Injection flaws
- Data validation
- Insecure storage

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data of other tenants.

Availability
These assessments test and validate the availability of the SaaS vendor.

- Authentication weaknesses
- Session management weaknesses

Many applications provide safeguards to automatically lock user accounts after successive incorrect credentials. However, incorrect configuration and implementation of such features can be used by malicious users to mount denial of service attacks.

Backup
The following assessments test and validate the security of the data backup and recovery services provided by the SaaS vendor.

- Insecure storage
- Insecure configuration

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data stored in backups.

Identity Management and Sign-on Process
The following assessments test and validate the security of the identity management and sign-on process of the SaaS vendor.

- Authentication weakness analysis
- Insecure trust configuration

Any vulnerability detected during these tests can be exploited to take over user accounts and compromise sensitive data.

Network Vulnerability Assessment

Network VA helps validate the network/host security in the cloud used for deploying the SaaS application in a self hosted model.

SaaS Deployment Model
The following assessments help test and validate the security of the infrastructure used to deploy the SaaS application.

- Host scanning
- Penetration testing
- Perimeter separation for dev/production systems
- Server hardening

- Firewall testing
- Router testing
- Domain name server testing
- Mail Server testing

The above assessments help ensure security of the SaaS deployment against external penetration and breaches and prevent loss of sensitive data.

Availability
The following assessment helps test and validate the availability of the infrastructure used to deploy the SaaS application.

- DoS testing

The above assessment helps test and validate the resilience of the SaaS deployment to denial of service attacks and help ensure availability of the service to end users.

Conclusion

The Software as a Service (SaaS) model offers customers significant benefits, such as improved operational efficiency and reduced costs. However, to overcome customer concerns about application and data security, vendors must address these issues head-on.

When it comes down to it, most enterprises' security concerns are centered on the lack of control and visibility into how their data is stored and secured with SaaS vendors. There is a strong apprehension about insider breaches, along with vulnerabilities in the applications and systems' availability that could lead to loss of sensitive data and money. Such challenges can dissuade enterprises from adopting SaaS applications.

The adoption of SaaS security practices - secure product engineering, secure deployment, GRC audits and regular SaaS security assessment - is vital to securing SaaS solutions. These can help identify any security issues upfront and ensure the safety of the data. SaaS vendors will benefit from the improved security of the solution and third-party validation of their security in the form of shortened sales cycles, and reduced operational risk. These measures will help them better answer any sales and marketing queries about security and address customer concerns. Customers will further be benefitted and assured about the security of their sensitive data and have higher confidence in the SaaS vendor.

Thus, adoption of the above SaaS security strategies and regular SaaS security assessment can enable SaaS vendors to boost customer confidence in the security of their solution and enable its faster and wider adoption.

To learn more about SaaS, email us on enquiry@leosys.net

About Us

Leo Technosoft - Cloud Computing R&D Center for product development, IT services and infrastructure management

Leo Technosoft is a hybrid Software Product Development Company based in India, US and UK. We partner with organizations, empowering them to attain cost effective product development in SaaS environment.

Our prime focus is on reducing your company's expenses while delivering you quality services. We have a global presence with offices in India, USA and UK. Our international footprint allows us to meet our client's needs regardless of their location. This has made us a partner-of-choice for small and medium sized companies looking for on-time delivery and high quality product development solutions.

We specialize in outsourced product development and IT services providing our global clients value for their money to meet specific business needs.

Visit our website - http://www.leotechnosoft.net

Asia Headquarters'

INDIA
Leo TechnoSoft Pvt Ltd
201, Tower S4,
Phase II, Cyber City,
Magarpatta Township,
Hadapsar, Pune - 411 013

Our Presence

USA – Florida, Los Angeles

Europe – Belgium

Singapore

Reach us at: Email: enquiry@leosys.net

Tel:

India: +91-20-2689 9853 | USA: 407-965-5509

SaaS Tenant