NCS- eTHIC Application Source Code Security Certificate

eTHIC end-to-end Audit Life Cycle Automation

eTHIC is the risk based audit management software suite that banks and financial institutions need. The web-based application is capable of processing more than 5 million transactions per day. Over 50% public and private sector banking institutions and financial services companies in India trust eTHIC to keep their data secure and clean. The clientele includes State Bank of India, the largest banking and financial services company in India by assets.

At NCS we being at developing software for BFSI, especially those in the electronic commerce arena, have reputations to protect. The main objective behind software assurance is making sure that software does what it is supposed to do. Our eTHIC software assurance encompasses more than just what it is supposed to do. The same core idea-making software is deeply entwined with software reliability and software safety as well.

Security throughout the software development life cycle (SDLC)

We at NCS practice a more proactive approach that incorporates a well designed secure development life cycle, and includes appropriate tools, processes and training. This eliminates, mitigate or reduce the risk of harm from security vulnerabilities. Developers analyze security from the moment the first line of code is written. This assures that risk and development costs are reduced over time.

Web application development remains as such, formal processes and best practices for developing web software are used. Currently we follow a set of standard steps, which define each phase of software creation. These phases are collectively referred to as the SDLC. It has become apparent that lack of security was a serious issue and also the most vital missing piece in the development process. In fact, security assurance in the past was relegated to the QA phase of development when it was considered at all. Now, however, forward-thinking organizations like NCS are adding security activities to every phase of the SDLC in order to discover flaws earlier and to significantly increase the security of the applications that are in production.

Typical security activities in each phase of the SDLC are as follows:

TRAINING:

Everyone involved in web application development were provided basic security training. Scalability and repeatability were critical aspects of effective security training programs.

REQUIREMENTS:

As software requirements are defined, the corresponding security requirements were also be defined. For example, if sensitive customer data is to be collected and stored, requirements on how the data should be encrypted, both in transit and at rest should were established as a requirement.

DESIGN :

Once the application requirements are captured, architecture is designed to incorporate all the software requirements. At this stage of development, necessary security controls are identified and included as part of the application.

Refer Annexure1

IMPLEMENTATION:

After requirements have been determined and an architectural design is in place, software development begins. Developers receive security feedback while they are coding. This feedback begins as early and as often as possible. Because this phase is often the most labor-intensive, continuously running automated security assessments were performed, which allows a developer to address issues in near-real time. This allows NCS to develop applications that are designed to be secure, rather than develop risky code; with security added as an afterthought.

QUALITY ASSURANCE:

The customized code is before it goes into a production environment, to ensure that the code behaves as expected. While most organizations currently test applications to ensure that the functional requirements are being met; We at NCS test if the application is secure, based on the security requirements.

PRODUCTION :

In the deployment phase, continual testing is vital to maintain security assurance and to protect against common application vulnerabilities. In addition, updates to applications that are already in production can introduce new flaws. Therefore, all code updates are subjected to source, QA and production testing.

Annexure-1

We, at NCS follow the below mentioned Security Practices for eTHIC Application Development/Customization.

1. Do not store unencrypted sensitive information on the client side
2. Properly encode or escape output
3. Code to prevent code injection
4. Code to prevent LDAP injection
5. Code to prevent arbitrary file upload
6. Do not use the clone() method to copy untrusted method parameters
7. Do not use Object. Equals() to compare cryptographic keys
8. Limit the lifetime of sensitive data
9. Do not use insecure or weak cryptographic algorithms
10. Store passwords using a hash function/in encrypted format
11. Avoid granting excess privileges
12. Define custom security permissions for fine-grained security
13. Ensure that security-sensitive methods are called with validated arguments