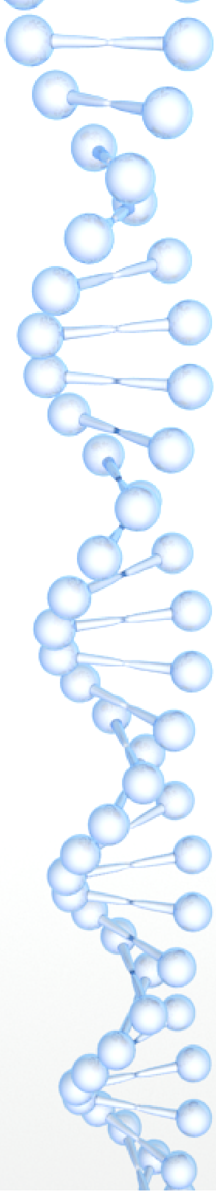
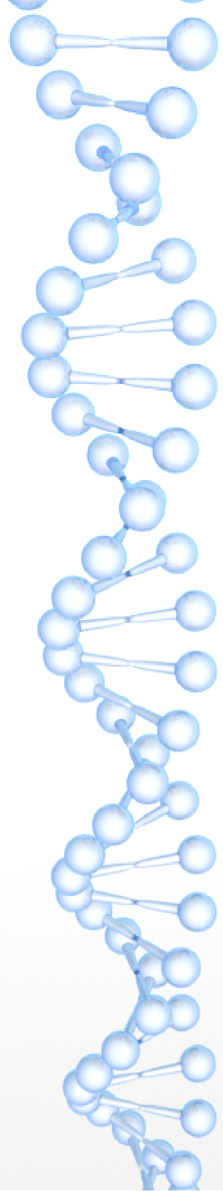


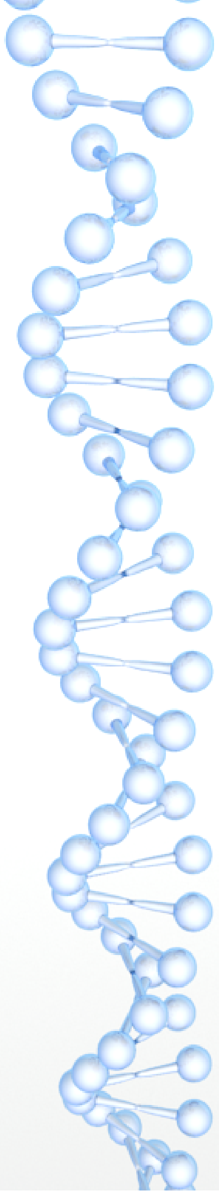
CCSP – Chapter 5

Understanding the Software Development Lifecycle (SDLC)



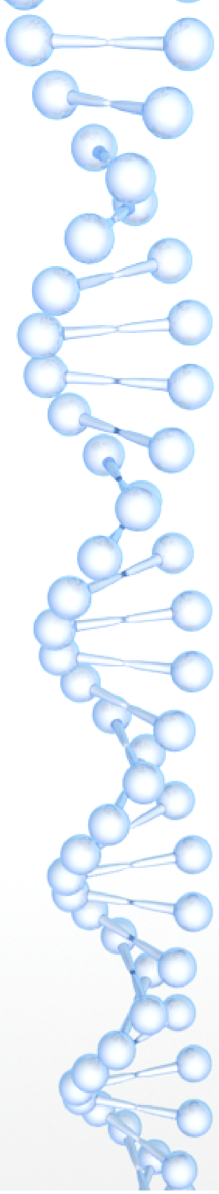
SDLC Phases

- High Level Requirements and Feasibility
- Testing
- Maintenance
- Business Requirements
  - Seems to be out of order?



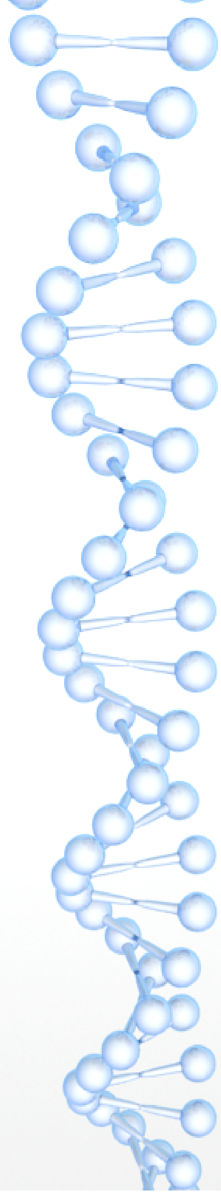
High Level Requirements and Feasibility

- High Level Aspects
  - Inputs
  - Outputs
  - Timing
  - Goals
  - Duration
  - Security
  - Costs
  - Benefits from current system
  - Value Add
    - Upgrade?
    - Replace?



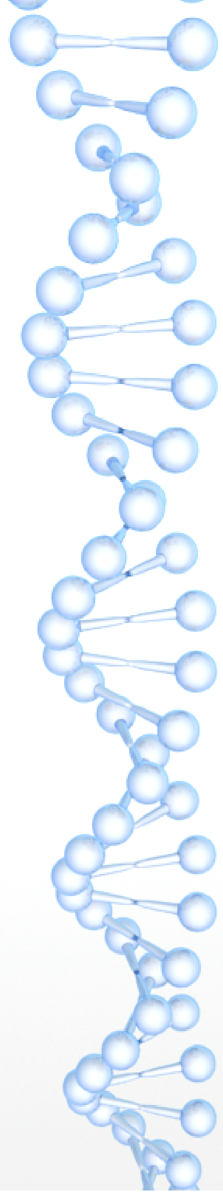
High Level Requirements and Feasibility

- Requirements Analysis
  - Goal – Translate high level requirements into a plan
    - Features/Functionality
    - Hardware/Software
- Design
  - Translate the plan into a structure that can be coded
  - Formal Security Requirements added
    - Threats/Mitigation/Minimization
- Development/Coding
  - Break down into coding components
    - Testing done on components
  - Longest Phase
  - Security static scanning / White box testing should be done as soon as coding starts
    - Typically delayed until end adding delays and complex fixes



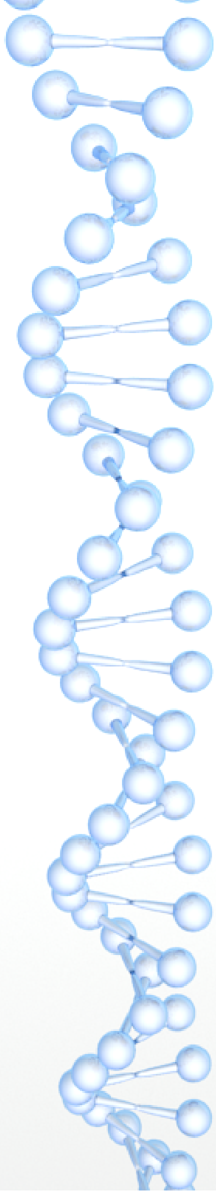
Testing

- Test plan development
  - Application Inputs/Outputs
  - Management approval
  - Resources made available to meet timeline
    - Personnel
    - System/Infrastructure
  - Include security scans
  - Validate code syntax for error and problems
    - Code quality checks
  - Output is a detailed test report



Maintenance

- Updates
  - Added Features
  - Bug fixes
  - Security patches
- Iterative process that will repeat the SDLC cycle



Business Requirements

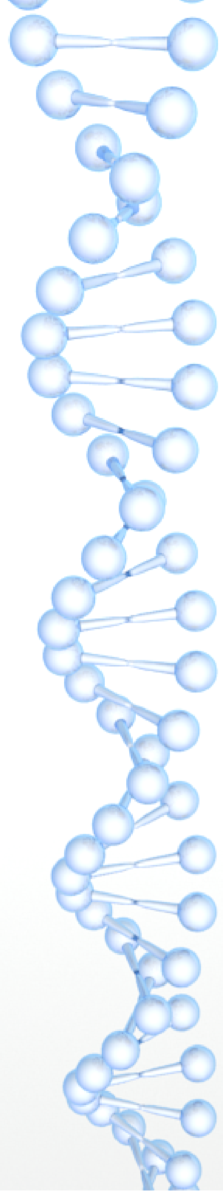
- Requirements target the business requirements
- Input from stakeholders
  - Better chance of gathering and understanding all the requirements
  - Should include end users
- Determine critical success factors
  - Should be measurable if possible

Business Requirements (cont)

- Software configuration management and versioning
  - Critical to ensure unity and cohesiveness
- Cloud differs from traditional
  - No patching but new images
- Automation is key
  - Puppet or Chef
- Many tools available (like GitHub)

Applying the SDLC

- Cloud Specific risks
- Quality of Service (QoS)

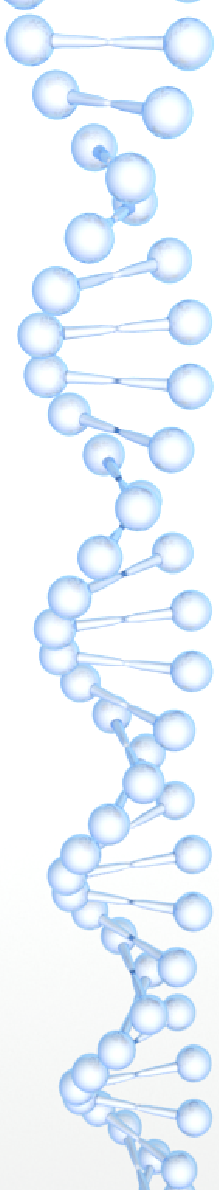


Cloud Specific risks

- Cloud Security Alliance (CSA)
  - Publishes the Treacherous 12
  - Covers risks specific to a cloud-based app
    - OWASP Top 10 for the Cloud

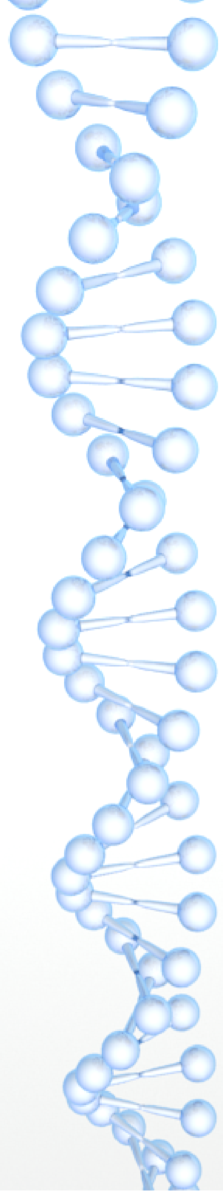
Cloud Specific risks – Treacherous 12

1. Data Breaches
    1. Any vulnerability can lead to this
    2. Possible to expose other application data hosted on same cloud if segmentation/isolation is not in place
    3. Magnified if hypervisor or management plan is compromised
2. Insufficient Identity, Credential and Access Management
    1. Can lead to other issues especially data breach or malicious inside vulnerabilities
3. Insecure Interfaces and APIs
    1. Heavy reliance on API's for automation and operations
    2. Risks to infrastructure and application
4. System Vulnerabilities
    1. Defects in the application-hosting framework
    2. Includes programming libraries and run-time environments



Cloud Specific risks – Treacherous 12 (cont)

5. Account Hijacking
    1. NO ACCOUNT SHARING!!
    2. Accounts based on user needs
    3. Multifactor authentication in place
6. Malicious Insiders
    1. Authorized user who uses access for unauthorized purposes
    2. Monitoring and auditing on sensitive systems should be in place
7. Advanced Persistent Threats
    1. Long-term program or malware to operate or steal data
    2. User education and training can help prevent malicious apps
8. Data Loss
    1. Can be purposeful or accidental deletion of backups, natural disasters, or encryption keys loss
    2. Can be an internal user or external compromise

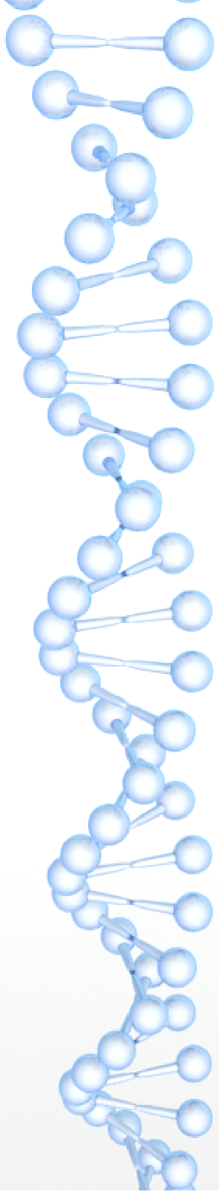


Cloud Specific risks – Treacherous 12 (cont)

9. Insufficient Due Diligence
    1. Responsibility of management, security, and users
    2. Training, policies, and enforcement mechanisms help
10. Abuse and Nefarious Use of Cloud Services
    1. Bypassing security controls
    2. Attacker can use to attack other apps or services
11. Denial of Service
    1. Too much traffic for app to deal with
    2. Can impact other tenants on the same provider
12. Shared Technology Issues
    1. Since all users share resources security issues impact everyone
    2. Cloud provider should layer additional security and resource monitoring

Quality of Service (QoS)

- SOA (Service-Oriented Architecture) is focused on monitoring and management of the business level and IT systems level
- Business Level
  - Measuring/monitoring of events, processes, KPI's (Key Performance Indicators).
- IT Systems
  - Security and system health, applications, services, networking, and any other IT infrastructure metrics

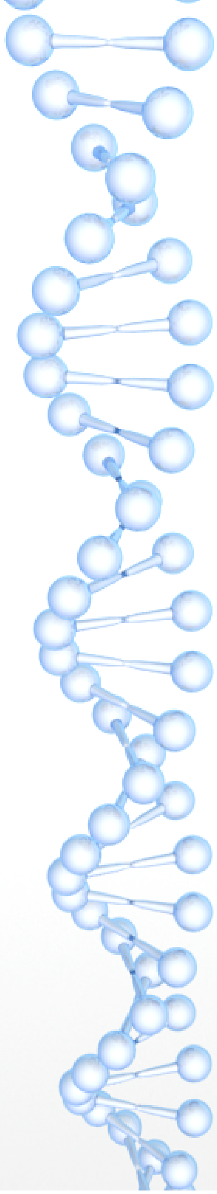


Threat Modeling

- STRIDE
  - Spoofing Identity, Tampering with data, Repudiation, Information disclosure, Denial of service (D.o.S), Elevation of privilege
  - KNOW THESE FOR THE EXAM
- DREAD
  - Damage Potential, Reproducibility, Exploitability, Affected users, Discoverability
  - Algorithm
    - Risk_DREAD = (Damage Potential + Reproducibility + Exploitability + Affected users + Discoverability) / 5

Cloud Application Architecture

- Other tools/technologies can be used for higher levels of security
- Follow the defense in depth philosophy
- Available tools/technologies
  - Supplemental security devices
    - Provide the layers of security and protection for the app
    - Provides better security by operating on different layers
  - Firewalls
    - Provide security starting at the perimeter
    - Should operate at the presentation, applicatons, and data layers
    - Virtual firewalls on the norm in the cloud environment
  - Web Application Firewall (WAF)
    - Filter HTTP traffic for things like SQL injection and cross-site scripting
    - Constant tuning of rules may be required
    - NOT a replacement for proper security controls



XML Appliances

- Used to consume, manipulate, accelerate and/or secure XML traffic
  - Commonly used to validate incoming XML
  - Typically put between the firewall and app server
  - Can control what users or apps and access the XML interfaces
  - Accelerators offload processing of XML data from the app and systems
  - Broker communication between cloud services and enterprise app

Cryptography

- Open/multitenant environment makes this critical
- Encryption of data at rest can occur in many places
  - Entire instance, storage volume, file, directory, or entire virtual machine
- Encryption of data in transit SSL, TLS, and VPN's are used
  - SSL is phased out and often will not meet requirements

Sandboxing

- The segregation/isolation of information or processes from within the same system or application
  - Can isolate data between users or communities
  - Sometimes needed for legal or regulatory requirements
- Also used for testing new code

Application Virtualization

- Software implementation to allow apps and programs to run in an isolated environment
- Very useful for testing
  - Testing/upgrading can be tested in a non-prod environment
  - Can test on different operating systems
- Not suitable for all applications
  - Issues with access to system drivers or hardware
  - Maintaining licenses

Identity and Access Management (IAM)

- Federated Identity
  - Allow trust access and verification cross multiple organizations
    - Each organization maintains its own identity and verification system
  - Must have an accepted standard and means to communicate with each other
    - SAML
    - OAuth
    - OpenID
    - WS-Federation

SAML

- **S**ecurity **A**ssertion **M**arkup **L**anguage
- XML-based markup language for security assertions
- Used for the information exchange between identity providers and service providers
- Flow when an entity authenticates through an identity provider
  1) SAML assertion is sent to the service provider containing all the required information
  2) Service provider determines identity, access level, and any other information or attributes about the entity

OAuth

- “The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf.”
  - https://tools.ietf.org/html/rfc6749
- An open protocol to allow secure authorization in a simple and standard method from web, mobile and desktop applications.

OpenID

- Based on Oauth 2.0
- Provides developers with an easy, flexible mechanism for authentication across organizations utilizing external identity providers
- Eliminates password stores and systems
- Uses browser tie-in which provides web-based applications an authentication mechanism which is independent of clients or devices

WS-Federation

- This specification defines mechanisms to allow different security realms to federate, such that authorized access to resources managed in one realm can be provided to security principals whose identities and attributes are managed in other realms. This includes mechanisms for brokering of identity, attribute, authentication and authorization assertions between realms, and privacy of federated claims.

  By using the XML, SOAP and WSDL extensibility models, the WS-* specifications are designed to be composed with each other to provide a rich Web services environment. WS-Federation by itself does not provide a complete security solution for Web services. WS-Federation is a building block that is used in conjunction with other Web service, transport, and application-specific protocols to accommodate a wide variety of security models.

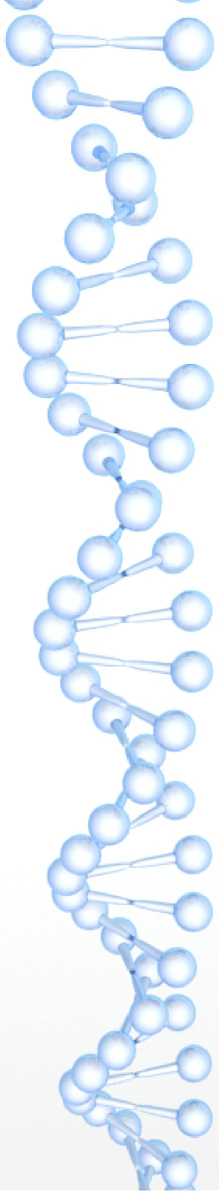
  - http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.html

Identity Providers (IdP)

- Holds authentication mechanisms for its users to prove their identity to the system at an acceptable level of security (authenticate)
- Once authenticated, the IdP can assert to other systems, service providers, or other parties the identity of the user.
- Can be a simple authentication success or info on the user.
- The Relying Party (RP) takes the assertion and uses it to grant access and determine the type and level access.
- The IdP and RP work together to facilitate authentication and authorization

Single Sign-On

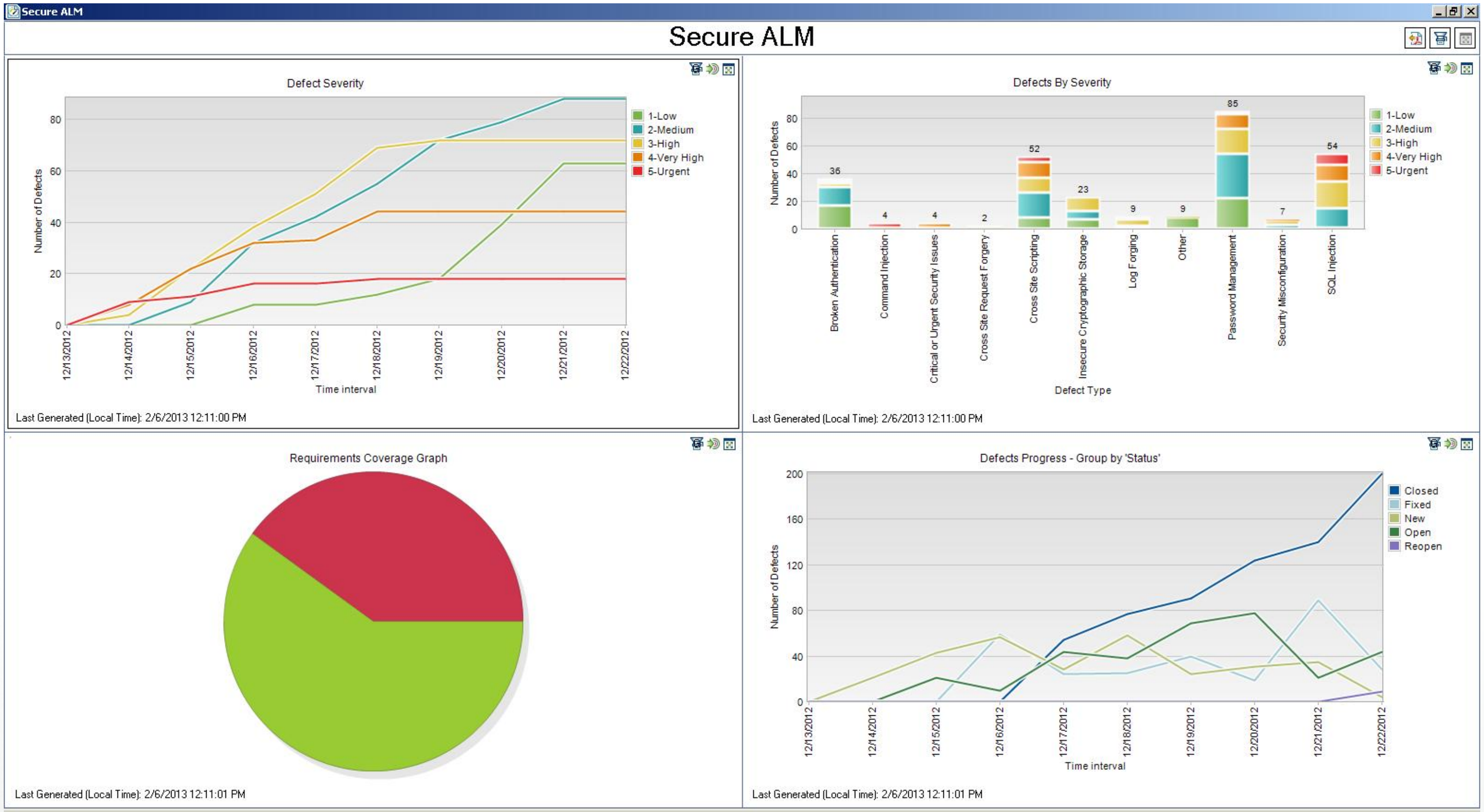
- Allows an entity to authenticate once at a centralized location
- Offloads systems and applications providing their own authentication.
- Systems and applications can then provide authorization based on the authentication via tokens



Multifactor Authentication

- Exceeds traditional authentication of user/password to include a second factor
- Three main components
  1) Something you know
     1) Username/password
  2) Something you have
     1) Thumb drive, RFID, RSA token, etc.
  3) Something you are
     1) Biometrics (fingerprints, iris scans, facial, etc)

SDLC Tools (ALM)



Secure ALM

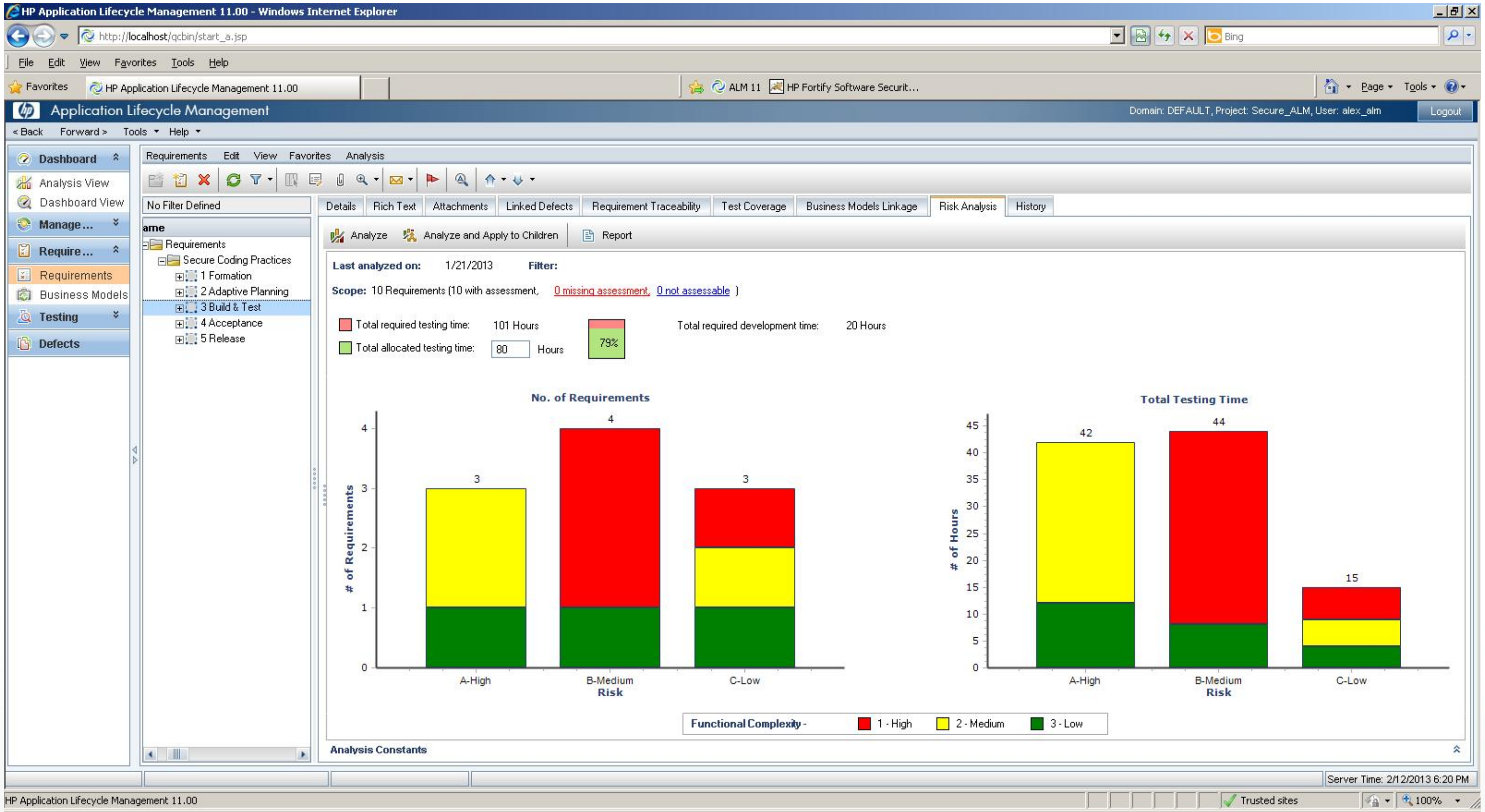
Last Generated (Local Time): 2/6/2013 12:11:00 PM

Last Generated (Local Time): 2/6/2013 12:11:00 PM

Last Generated (Local Time): 2/6/2013 12:11:01 PM

Last Generated (Local Time): 2/6/2013 12:11:01 PM

SDLC Tools (ALM)



Domain: DEFAULT, Project: Secure_ALM, User: alex_alm

Last analyzed on: 1/21/2013 **Filter:**

Scope: 10 Requirements (10 with assessment, 0 missing assessment, 0 not assessable)

🟥 Total required testing time: 101 Hours

🟩 Total allocated testing time: 80 Hours

79%

Total required development time: 20 Hours

Server Time: 2/12/2013 6:20 PM