Checklist for Securing Application Design			
	Category	**Vulnerable Area**	**Facts to ANALYSE**
Design	**Code Flow – Division of code based on MVC**	**Presence of backdoor parameters/functions/files**	1. Are there backdoor/unexposed business logic classes?
			2. Are there unused configurations related to business logic?
			3. If request parameters are used to identify business logic methods, is there a proper mapping of user privileges and methods/actions allowed to them?
		Placement of checks	1. Are security checks placed before processing inputs?
		Insecure Data Binding Mechanism	1. Check if unexposed instance variables are present in form objects that get bound to user inputs. If present, check if they have default values.
			2. Check if unexposed instance variables present in form objects that get bound to user inputs. If present, check if they get initialized before form binding.
	Authentication and Access Control Mechanism	**Insecure authentication and access control logic**	1. Is the placement of authentication and authorization check correct?
			2. Is there execution stopped/terminated after for invalid request? I.e. when authentication/authorization check fails?
			3. Are the checks correct implemented? Is there any backdoor parameter?
			4. Is the check applied on all the required files and folder within web root directory?
		Redundant configuration	1. Is there any default configuration like Access- ALL?
			2. Does the configuration get applied to all files and users?
			3. Incase of container managed authentication - Is the authentication based on web methods only?
			4. Incase of container managed authentication - Does the authentication get applied on all resources?
		Insecure Session management	1. Does the design handle sessions securely?
		Weak Password Handling	1. Is Password Complexity Check enforced on the password?
			2. Is password stored in an encrypted format?
			3. Is password disclosed to user/written to a file/logs/console?
	Data Access Mechanism	**Presence of sensitive data in configuration/code files**	1. Are database credentials stored in an encrypted format?
		Presence/support for different insecure data sources and their related flaws	1. Does the design support weak datastores like flat files
	Centralized Validation and Interceptors	**Weakness in any existing security control**	1. Does the centralized validation get applied to all requests and all the inputs?
			2. Does the centralized validation check block all the special characters?
			3. Does are there any special kind of request skipped from validation?
			4. Does the design maintain any exclusion list for parameters or features from being validated?
Architecture	**Entry Points**	**Insecure Data handling and validation**	1. Are all the untrusted inputs validated?
	External Integrations	**Insecure data transmission**	1. Is the data sent on encrypted channel? Does the application use HTTPClient for making external connections?
			2. Does the design involve session sharing between components/modules? Is session validated correctly on both ends?
		Elevated privilege levels	1. Does the design use any elevated OS/system privileges for external connections/commands?
Configuration	**External API's used**	**Known flaws present in 3rd party APIs/functions**	1. Is there any known flaw in API's/Technology used? For eg: DWR
	Inbuilt Security Controls	**Common Security Controls**	1. Does the design framework provide any inbuilt security control? Like <%: %> in ASP.NET MVC.
			2. Are are there any flaw/weakness in the existing inbuilt control?
			3. Are all security setting enabled in the design?

Ashish Rao
Ph: 91-9819080470
rao.ashish20@gmail.com
Blog: http://artechtalks.blogspot.in/