**WEB安全点和推荐实现方式**

**SQL注入：**推荐使用框架整体实现，mybatis和jdbc都可以，但建议mybatis，测试方便，一分钟就可以覆盖测试完。

**XSS注入：**输入过滤特殊字符（&、<、>、"、'、(、)七种特殊字符），输出统一html编码；

**上传下载：**

上传：

1、文件扩展名白名单机制（如果是上传压缩文件解压，需要校验解压后每个文件的类型）

2、判断上传文件类型（后缀+二进制头）

3、文件自动重命名后保存

4、服务器端控制文件保存路径

5、对用户可以控制的目录名称过滤（不要用用户输入的目录和文件名）

6、检查上传文件大小

7、上传下载附件权限加到祥云权限控制列表

8、CSRF Token避免上传的DOS攻击

下载：

需要验证下载权限！

下载还不能跨目录，只能下载指定文件。

**接口安全：**

**认证鉴权：**所有接口默认都要提供鉴权（外部接口Token/秘钥/口令鉴权，内部接口HMAC鉴权，所有参数需要纳入HMAC签名范围内）。所有请求参数无法横向越权，所有鉴权分支在判定时需要考虑伪造篡改的场景，只有Token，秘钥这种认证无法篡改，不推荐任何场景下使用IP白名单等其它认证方式，都可以绕过。

**DOS拒绝服务：**请求中有数组的，或者序列化数据的，后台存在循环操作的场景，需要排查是否做校验，限制参数个数，避免DOS；

**文件IO操作：**如果接口涉及到文件操作，需要关注请求中是否有携带path参数，避免产生目录遍历后带来的任意文件读取，任意文件删除等漏洞！

**通用检查点：**

1.    明文的秘钥不允许出现在系统任何地方（红线），包括配置文件，日志打印，存储和传输；

2.    任何接口默认需要鉴权，内部网元之间HMAC校验消息完整性，HMAC需要涵盖POST/GET提交的所有业务参数；

3.    输出到WEB页面的信息需要做XSS过滤，也就是对红线要求的字符进行HTML编码；

4.    所有配套的SDK（比如DCS,SDS,DMQ等）建议升级到最新，因为老版本存在很多安全问题；

5.    传输默认走HTTPS，HTTP需要考虑是否含有敏感信息和个人数据，以及篡改消息带来的风险；

6.    算法目前AESCBC，HMACSHA256，PBKDF2，RSA2048，SHA256基本可以满足需要，其它算法有用到一起讨论是否合法；

1.进程启动用户检查；

2.权限加固，用户下umask 077，所有文件other组权限检查；

3.配置文件重点检查，清理废弃配置文件，以及注释的敏感信息配置项；

4.梳理各部件对外网元的方式，默认https，http通信的话确认是否有消息认证机制，以及是否包含敏感信息；

5.场景上再梳理下，所有接口只能操作本用户的数据，横向越权现在是ICSL的重点；

6.所有敏感信息需要加密保存，系统中任何地方不能有明文的秘钥信息，token信息，密码信息等；

7.HTTPS场景，证书需要SHA256RSA，TLS1.1 1.2，秘钥加密保存；