企业安全漏洞综合治理方案

目录

一、 引言

当今数字化时代，企业面临着日益增多和复杂化的安全漏洞问题，这些安全漏洞可能导致数据泄露、系统瘫痪、商业机密被窃取等严重后果，对企业的运营和数据安全构成了严重威胁。

根据最新的数据统计显示，全球每天有数十万次的安全漏洞攻击事件发生，其中一半以上是由未修复的已知漏洞引起的，这些攻击可能导致企业损失数百万甚至数亿的资金，从而损坏企业品牌和声誉，甚至使企业面临政府机关或客户的问责或法律诉讼。

因此，对已知安全漏洞的修复是企业在安全建设工作中不可或缺的一项重要任务。只有及时修复漏洞，加强系统和应用程序的安全性，企业才能保护自身的核心业务运作的可持续发展。

综上，通过对已知安全漏洞的修复对于企业发展有以下意义：

1. 及时有效的安全漏洞修复是企业可信度和声誉的重要基石；
2. 及时的安全漏洞修复可以降低企业潜在的经济损失；
3. 积极的安全漏洞修复可以解决企业面临的法律和合规风险。

二、 安全漏洞识别与评估

企业面临的安全漏洞的形式多种多样，包括但不限于软件缺陷、配置错误、系统漏洞，企业进行漏洞修复的前提是：必须首先对这些漏洞进行识别和评估，识别、掌握、记录安全漏洞的信息，才能在该基础上开展漏洞修复的工作。

企业识别安全漏洞的方式有很多，以下是常见的安全漏洞识别方法：

1. 使用漏洞扫描工具：使用自动化漏洞扫描工具，如 Nessus、AWVS、OpenVAS、Nmap 等，对企业的网络、应用程序和系统进行安全漏洞扫描，自动化工具内置了发现常见安全漏洞的检测方法或功能，可以发现已知的安全漏洞或明显的技术类型的安全漏洞，并可以生成相应的漏洞扫描报告。

2. 源代码安全审计：通过工具和人工审计应用程序的源代码，从应用程序

实现维度识别代码中的潜在漏洞和安全弱点。代码审查可以手动进行，也可以使用静态代码分析工具，如 Fortify、Checkmarx 等，以辅助识别代码中的安全问题。

3. 渗透测试：通过模拟真实攻击场景，对企业的网络、系统和应用程序进行主动安全测试，通过测试人员的经验和能力，发现潜在的漏洞和安全风险，渗透测试是由专业的安全人员、团队或公司进行，过程中往往也会使用安全漏洞扫描工具或其他安全测试工具，如 Metasploit、Burp Suite 等。

4. 使用软件成分分析工具/系统：分析应用程序源代码，提取项目依赖的第三方组件及其版本、许可证等信息，生成软件物料清单（SBOM，Software Bill of Materials），根据 SBOM 分析项目是否使用了存在已知漏洞的开源组件，从而解决由于组件安全漏洞引起的安全风险，同时，一旦发生组件类安全漏洞，也可以根据 SBOM 快速排查受影响项目。

5. 攻防演练：通过真实的攻防对抗演练，组织内部或第三方安全团队对企业进行安全攻击，从攻击者视角发现企业的安全漏洞和弱点，同时可以帮助企业评估自身的安全防御能力，及时修复潜在的漏洞。

6. 网络流量分析：通过分析网络流量，识别异常的网络行为，从而发现可能的攻击行为，进而定位企业的安全漏洞，比如使用态势感知平台、Wireshark、Snort 等监测和分析网络流量，识别潜在的安全漏洞和威胁。

7. 订阅安全漏洞通告：订阅漏洞和安全通告，如 CVE（Common Vulnerabilities and Exposures）、厂商公告等，及时了解最新的安全漏洞和安全威胁信息，并结合企业资产信息及时进行安全漏洞排查，快速处理相关的安全漏洞和威胁。

企业发现安全漏洞的方式和渠道有很多，除以上方法外，不同维度和不同场景下都有不同安全漏洞识别方法和手段，但最终目标都是识别企业自身的安全漏洞。

三、 安全漏洞修复策略

企业在进行漏洞修复时，应建立一套相应的漏洞修复策略，漏洞修复策略包含以下六个因素：

1. 团队构建和合作；
2. 漏洞修复优先级；
3. 漏洞修复成本；
4. 修复时间分配；
5. 修复资源分配；
6. 持续监控与反馈策略；

（一） 团队构建和合作

在进行漏洞修复前首先需要建立一个漏洞评估和修复团队，一般为临时组成，临时性取决于整体漏洞修复完成的周期。团队一般由安全、研发、运维及相关负责人组成，团队负责人由企业业务负责人或管理层担任，其中安全人员可能涉及第三方人员，如企业无安全团队，则需要第三方安全服务团队承担安全角色。团队成员分别负责如下内容：

- 安全人员：主要负责提供安全技术支持，如漏洞分析评估、修复指导、漏洞修复验证等工作；
- 研发人员：主要支持应用相关漏洞的修复工作；
- 运维人员：主要负责服务器、网络设备、数据库、安全设备（部分企业安全设备由运维负责）等的系统升级、补丁下发、策略配置、安全加固相关工作；
- 团队负责人：负责人员、资源的协调，包括各个岗位负责人的沟通、协调；

为了保证安全漏洞修复工作能够顺利进行，需要执行以下工作来维持漏洞修复团队的正常：

1) 沟通与协调

团队成员之间需要进行及时、清晰的沟通，确保每个人都了解修复计划、进展和任务分配，确保大家在同一个目标上保持一致，因此沟通与协调是必不可少

的。

在实际执行中，需要团队建立一个沟通渠道，如群聊，渠道成员应该包含漏洞修复需要的所有成员。另外安全人员如果是来自第三方安全团队，企业应设立一个接口人，用于与安全人员的对接，避免因为安全人员与团队其他成员不熟悉，产生沟通障碍而影响漏洞修复进度及质量。

2) 角色分工

团队需要明确每个成员的职责和角色，确保每个人都知道自己应该做什么，这有助于避免任务重叠或者遗漏。

上文提到在安全人员为第三方安全团队时，企业应设立接口人，这个接口人可以认为是团队负责人的代言人，因为负责人处理的工作比较多时，无法时时处理漏洞修复工作，只能在接口人在遇到困难时，再由领导出面进行协调解决，因此接口人需要是一个能解决大部分问题（企业内部相关事项：跨部门沟通，资源协调等）的人员。

3) 信息共享

漏洞修复过程中，团队成员应该及时共享修复过程中发现的信息和漏洞详情。如漏洞修复人员应及时将发现的问题反馈至安全人员，安全人员及时进行分析，协助漏洞修复人员进行问题解决，这有助于提高整个团队对问题的认识，并促进共同解决问题。

另外，漏洞修复人员应对漏洞修复的修复方式进行记录，这样可以有以下三点好处：

a) 记录漏洞修复相关内容，确保漏洞修复工作有迹可寻，如果修复失败或影响到业务其他部分，可以回溯以及回滚修复内容；

b) 记录漏洞修复相关内容，方便安全人员在漏洞复测时进行查阅，确认漏洞修复是否完全；

c) 记录漏洞修复相关内容可以沉淀至企业内部安全漏洞修复经验库，以后再遇到类似问题，可以作为参考；

4) 知识共享

在漏洞修复过程中，鼓励团队成员分享他们的经验和知识，这有助于提高

整个团队的技术水平，并为将来的修复工作积累经验。如安全人员需详细的为漏洞修复人员描述漏洞原理、产生原因、漏洞危害及修复建议，而漏洞修复人员应将漏洞产生处的相关实现技术及现状进行描述，有助于安全人员协助选择最优的漏洞修复方法。

5）监控与评估

漏洞修复时，团队中的安全人员应该跟踪修复过程中的进展，并定期进行评估和审查，这有助于发现问题，并及时采取纠正措施。首先，评估漏洞修复是否会影响业务，如果会产生影响，那么是否是可以承受的，这是需要负责人与业务方进行沟通的。其次，漏洞修复完成需要重新评估漏洞是否修复完全，这需要安全人员进行专业的安全检测，以及业务人员的修复影响评估。最后，确认漏洞完成修复后，需要由业务方或系统相关监控人员观察系统是否存在异常。

6） 预案和备份计划

在进行安全漏洞修复前，团队需要根据企业实际情况协助制定相关预案和备份计划。在修复过程中，一般由运维团队按计划备份关键数据和系统配置，以防修复过程中出现问题导致数据丢失或系统故障。

（二） 漏洞修复优先级

漏洞修复团队建立后，安排好漏洞修复前的相关工作后，就要进入漏洞修复阶段。开展漏洞修复时，首先需要确认先修复哪些漏洞，后修复哪些漏洞，因此团队需要制定漏洞修复优先级。如果不进行漏洞优先级排序，很有可能出现一些危害程度较高、产生影响较大的漏洞在漏洞修复过程中被攻击者利用。因此，制定漏洞修复优先级就是对漏洞修复紧急程度进行分类，如紧急、高、中、低。

漏洞修复优先级可以参考以下方面：

1）漏洞严重性

对每个漏洞进行严重性评估，考虑其可能造成的影响和潜在风险。可以使用CVSS（Common Vulnerability Scoring System）等评估标准，根据漏洞的攻击复杂性、影响范围和可利用性等因素进行评估。这项工作由安全人员参考 CVSS 等标准进行，常见的漏洞扫描器也会根据扫描结果直接给予相应的CVSS评分。

2）业务影响程度

评估漏洞对企业业务的潜在影响。考虑漏洞可能对业务连续性、数据完整性和客户信任等方面产生的影响。根据业务关键性和敏感性，对漏洞进行优先级排序。对业务影响程度这块一般由业务相关人员进行分析即可，但是在实际场景中，一般会根据资产的重要性来决定业务影响评分。

比如同样一个漏洞，A 是边缘资产，B 是核心资产，那么这个漏洞在资产 B 的修复优先级就应该更高。具体执行时，需要先将资产进行归类、分级，然后通过将漏洞严重性与资产级别相结合判断修复优先级。

3）攻击概率评估

评估漏洞被利用的概率和风险。考虑漏洞的公开程度、已有的攻击利用情况和攻击者的动机，以判断漏洞被利用的可能性和影响程度。攻击风险评估工作需要专业的安全人员进行，因为这里需要对公开可利用安全漏洞、安全经验的积累，才能确认众多漏洞中哪些是可被利用的，哪些是暂无公开利用方式的。

4）漏洞修复复杂性评估

评估修复漏洞的复杂性和工作量。考虑修复所需的技术难度、资源投入和可能的影响范围。较复杂和耗时较长的修复工作可以优先考虑，以便及早解决潜在的问题。这项评估比较简单的依据是漏洞类别，及是否涉及系统重启等内容进行评估。如：核心资产服务器系统存在漏洞，且需要重启服务器才能完成修复，那么就属于比较复杂的漏洞，这种情况漏洞修复复杂的评分就相对较高。

5）业务需求和合规要求

考虑业务需求和合规性要求对修复优先级的影响。根据行业标准、法规和合规性要求，将修复与业务和合规性目标相结合，制定相应的优先级。本项一般适用于监管单位下发的漏洞，以及不修复将违反行业标准、法规合规项要求等情况（如：App 隐私不合规需要整改），这时就需要将优先级调高。

（三） 漏洞修复成本

确定漏洞修复优先级之后，和在投入资源进行漏洞修复之前，需要对漏洞修复的成本进行评估。评估的目的是为了平衡修复成本和安全风险，实现安全工作

的成本效益最大化。漏洞修复成本评估有助于修复团队确定哪些漏洞值得修复，哪些漏洞不值得修复，以及修复漏洞企业所要付出的代价有哪些，方便团队负责人协调企业资源进行修复工作，并就修复工作向上汇报，获得企业管理层的认可。如果修复漏洞的代价过于昂贵，企业管理层在没有确认或授权投入资源进行修复时，可能会让漏洞修复或安全团队陷入两难的境地。

- 漏洞修复成本的评估包括以下方面：
- 是否影响运营环境或基础环境的操作；
- 是否会影响到业务的正常运行，如果是，影响的业务损失预计是多少；
- 修复工作需要投入多少一次性资源，如过渡的计算资源；
- 修复工作需要投入什么岗位的人员，多少人，执行多长时间的工作；
- 修复工作是否涉及公司其他部门的协助，以及协助哪些内容，多长时间；
- 修复工作是否需要引入外部资源，预计的费用是多少；

根据修复成本的评估，以及每一项安全漏洞的利用概率、漏洞利用的预计损失，计算单一漏洞和全部漏洞修复的成本，以及修复方案执行后的漏洞利用概率，从而评估每一个漏洞修复的必要性，以确保投入修复漏洞资源的有效性，即：

漏洞数量*（修复前漏洞利用概率*漏洞利用预计损失）-漏洞数量*（修复后漏洞利用概率*漏洞利用预计损失）>漏洞修复成本

（四） 修复时间分配

假如漏洞修复优先级分为 4 级，分别是紧急、高、中、低，那么不同优先级的漏洞修复时间也应该是不同的，紧急漏洞的修复时间必须是最短的，高则次之，中则第三，低则最慢。另外，不同修复优先级的漏洞，不同类型（应用漏洞、主机漏洞、数据库等）的漏洞的修复时间点安排也是非常重要的。

1） 修复时间窗口规划

根据漏洞资产分布情况，对业务影响情况、漏洞修复复杂度结合漏洞修复优先级来决定漏洞修复时间窗口，可以与业务团队、运维团队和其他相关方进行沟通，确保修复窗口的安排符合整体运营和维护的需求，确保各方对修复时间窗口表的了解和支持。

这里列举一个漏洞修复时间窗口的安排：1.核心资产漏洞、需要重启服务/主机,则尽可能安排在业务低峰期进行修复;2.边缘资产漏洞,对业务影响较小，但漏洞被利用，可能直接导致主机被攻击者控制，则第一时间进行修复。因此在具体实施时，仍需参考漏洞危害情况、漏洞资产分布情况、漏洞修复复杂度等因素进行决定。

2） 确定漏洞修复窗口

考虑到漏洞的严重性和修复的复杂性，越是优先级高的安全漏洞，修复窗口越短，其中低危漏洞往往不做强制性修复要求。

常见的漏洞修复窗口如下：

漏洞修复优先级	漏洞修复窗口（天）
严重	1
高	1-3
中	3-7
低	15-20

3） 定期评估和优化

企业需要定期评估安全漏洞修复的时间管理策略，并根据实际情况进行优化。根据经验教训和漏洞修复的效果，调整时间管理策略，提高修复效率和质量。

（五） 修复资源分配

企业在安全漏洞修复前，我们需要进行人员、资源需求评估，在安全漏洞修复中，我们需要根据实际情况对现有资源进行动态调整，在安全漏洞修复后，我们需要做协调资源好监控。合理的资源分配是确保漏洞修复工作能够高效进行的关键之一。

主要参考如下内容进行资源分配：

1） 修复工作量评估

如上文所述，对每个修复任务评估所需的工作量和资源投入，需考虑修复的复杂性、涉及的系统和应用程序数量，以及所需的人力和技术资源。

2） 资源合理分配

根据漏洞修复优先级和工作量评估，优先将资源分配给高优先级的修复任务，确保漏洞修复优先级高的漏洞威胁能在第一实际得到解决。

3） 优化修复流程

优化修复流程，以提高资源利用效率。确保流程简洁、清晰，并避免重复劳动或无效的工作。如果企业拥有一些自动化补丁管理工具，我们则可以利用其快速解决一些漏洞，从而可以减少人工操作，节约大量修复时间。

4） 外部资源合作

当企业在没有专业的安全团队，无法高效、甚至无法完成漏洞修复工作时，应该考虑与第三方合作进行安全漏洞修复。专业的安全团队可以提供相应的安全资源和专业的安全知识，帮助企业高效地进行安全漏洞修复工作。

5） 优先级调整和动态分配

在漏洞修复过程中需要根据修复进度和实际情况，及时调整修复任务的优先级和资源分配。在修复过程中，可能会发现新的漏洞或修复任务的优先级可能发生变化，因此需要灵活调整资源分配。

6） 监控和评估

在漏洞修复过程中需要监控资源分配的效果，并定期评估修复工作的进展和质量，以及资源分配的合理性和效率，根据评估结果进行必要的调整和优化。

例如:某研发团队在修复A业务系统的支付漏洞时,已经进行2次修复工作，但是经过安全团队检测后，发现仍然未完全修复漏洞，那这时就应该加大安全团队对该研发人员的技术支持，结合业务情况与研发人员对该漏洞进行分析，并贴合业务协助研发人员进行漏洞修复。

（六） 持续监控与反馈

在完成漏洞修复工作后，企业需要持续监控已修复的安全漏洞，并向安全团队及时反馈监控到的异常情况，一方面是保证漏洞修复措施的有效性，另一方面及时发现新产生的安全漏洞。企业也要定期进行漏洞检测工作（包括但不限于渗透测试、代码审计、开源组件风险扫描、漏洞扫描、攻防演练），及时发现新风险，并消除威胁。企业通过对修复漏洞的持续监控和反馈，可以及时发现和修复

新的漏洞，提高整体的安全性和抗攻击能力。

1) 安全漏洞跟踪：持续监控已修复的安全漏洞，确保修复措施的有效性。跟踪漏洞的状态和修复进展，以确保没有新的问题出现。
2) 安全事件响应：企业应及时响应新的安全事件和漏洞，需要建立安全响应团队或小组负责处理新发现的漏洞，并采取相应的措施进行修复。
3) 定期漏洞检测：定期对企业资产进行漏洞扫描、渗透测试、代码审计、开源组件风险扫描，甚至组织攻防演练行动来验证企业整体的安全及防御能力。
4) 安全漏洞报告：定期向企业管理层和相关利益相关者提供安全漏洞修复的报告，包括但不限于安全漏洞修复的进度、已修复的漏洞数量和修复效果评估。

四、 漏洞修复方案

（一） 漏洞修复流程

企业遵循漏洞修复流程，可以更加有序和高效的修复安全漏洞，提高整体的安全性和稳定性。以下是一个通用漏洞修复流程：

1) 修复计划制定：依据上文提到的安全漏洞修复策略来制定修复计划。先建立一个漏洞修复团队，其次将现有漏洞进行分类，划分出漏洞修复优先级，然后分配修复时间窗口和修复周期，接着根据实际情况合理分配人员及其它相关资源。
2) 修复措施实施：需要根据修复计划，实施相应的修复措施，包括代码修复、补丁安装、配置调整等操作。
3) 测试与验证：在实施漏洞修复措施后对修复措施进行测试和验证，确保修复的有效性和不会引入新的问题。
4) 部署与发布：漏洞修复人员需要根据计划中制定的时间节点将修复措施部署到生产环境，并进行发布，确保修复措施在整个系统中生效。

5) 监控与反馈：漏洞修复团队需要持续监控修复后的系统，确保修复的有效性和不会再次出现漏洞，及时反馈修复结果和相关报告。

6) 审查与总结：漏洞修复团队需要对修复过程进行审查和总结，评估修复策略的有效性和改进点，有助于提高漏洞修复的效率和质量。

（二） 漏洞修复方式

1）控制输入输出

通过控制输入输出可以修复大部分漏洞，如 SQL 注入漏洞、XSS（跨站点脚本攻击）漏洞等，因为恶意攻击者会通过输入将攻击代码输入应用程序，从而对应用程序发起攻击，如果应用程序在输入阶段做好验证、过滤，那么将使大部分攻击失去作用。输出作为应用系统在交互过程中给与用户的反馈，也应该做好相应的处理，如最小化权限输出、不直接输出应用程序错误信息。

在控制输入输出时，可参考以下操作：

a) 输入验证和过滤：对于用户输入，要进行严格的验证和过滤，确保只接受预期的输入，并拒绝不合法或潜在有害的输入。

b) 防止过度授权：只为用户提供必要的功能和权限，避免过度授权，以降低潜在的攻击面。

c) 输出编码和转义：在将数据输出到前端或其他系统时，确保对特殊字符进行正确的编码和转义，以防止跨站点脚本攻击（XSS）等问题。

d) 异常处理：正确处理异常情况，避免将敏感信息泄露给攻击者。

e) 输入输出日志记录：记录输入输出数据，以便在发生安全事件时进行溯源和调查。

以 SQL 注入漏洞为例：

漏洞名称	SQL 注入漏洞
漏洞类型	应用类型漏洞
漏洞危害	SQL 注入漏洞会直接泄露应用系统数据 SQL 注入漏洞可能导致系统数据被篡改

	SQL 注入漏洞可能导致服务器沦陷 .....
漏洞原理	SQL 注入是由于应用程序未对用户输入进行合法校验，程序员在编写 SQL 语句相关代码时，未采用安全编码，导致攻击者可以将恶意 SQL 语句拼接至应用的 SQL 语句中，从而在数据库中执行，以此达到攻击目的。
控制输入输出	程序员可以通过控制输入输出来修复 SQL 注入漏洞。如下为两种修复场景： 1.程序员可以通过控制用户输入的数据类型来修复 SQL 注入漏洞，如存在 SQL 注入漏洞的位置是手机号输入处，那么程序员通过判断输入的手机号是否为 11 位的手机号就可以直接将此处的 SQL 注入漏洞修复，因为攻击者需要输入 SQL 语句，而程序只接受 11 位的手机号，因此是可以修复 SQL 注入漏洞的。 2、程序员可以通过控制输出修复 SQL 注入漏洞，当应用程序存在 SQL 报错注入漏洞时，程序员可以通过异常信息处理，统一报错信息，不再将 SQL 错误信息展示至客户端，那么 SQL 报错注入漏洞就可以被修复，因为 SQL 报错注入漏洞是因为攻击者在使用一些特殊的函数或字符时，客户端会显示 SQL 错误信息，而攻击者就是利用 SQL 错误信息开展的 SQL 注入攻击，因此控制输出错误信息后，SQL 报错注入漏洞就会被修复。 修复 SQL 注入漏洞的方式有很多，控制输入输入只是其中一种方式，而且大部情况下是需要结合多种漏洞修复方式才能完全将漏洞修复。

2）使用安全编码

安全编码可以解决许多常见的安全漏洞，也是程序员在修复安全漏洞时常用的一种手段。其实安全编码的目标是通过正确的编码和实施安全措施来预防和减轻潜在的安全威胁。但是由于很多程序员由于经验不足，在编写程序时未采用安

全编码的方式进行程序编写，会直接产生漏洞，因此在被攻击者或安全人员发现后，在安全人员的帮助下或自我学习的情况下会通过安全编码的方式去解决安全漏洞。安全编码可以解决的漏洞包括但不限于以下几种：

a) 跨站脚本攻击（XSS）：通过正确的输入验证、输出编码和转义，可以防止恶意脚本注入到网页中，从而保护用户免受 XSS 攻击。

b) 跨站请求伪造（CSRF）：通过实施适当的 CSRF 令牌和验证机制，可以防止恶意网站或链接利用认证用户的身份执行未经授权的操作。

c) SQL 注入：通过使用参数化查询或预编译语句，可以防止恶意用户通过输入恶意 SQL 代码来篡改数据库或获取敏感数据。

d) 未经授权访问和越权访问：通过正确的访问控制和权限管理，可以限制用户的访问权限，防止未经授权的用户访问敏感信息或执行受限操作。

e) 文件包含漏洞：通过正确的文件路径验证和过滤，可以防止恶意用户通过包含恶意文件路径来执行任意代码。

f) 不安全的直接对象引用：通过正确的身份验证和授权机制，可以防止用户直接访问未经授权的对象或资源。

g) XML 外部实体（XXE）攻击：通过禁用外部实体解析或使用安全的 XML 解析器，可以防止恶意用户利用恶意实体进行信息泄露或拒绝服务攻击。

h) 敏感信息泄露：通过正确的输入验证、输出编码、加密和安全存储等方法，可以防止敏感信息（如密码、信用卡号等）在传输或存储过程中被泄露。

i) 不安全的重定向和转发：通过验证和过滤重定向和转发的目标地址，可以防止恶意用户利用不安全的重定向和转发机制进行钓鱼攻击或欺诈行为。

上文中我们通过控制输入输出可以修复 SQL 注入漏洞，这里我们使用安全编码的方式来修复 SQL 注入漏洞：

漏洞名称	SQL 注入漏洞
漏洞类型	应用类型漏洞
漏洞危害	SQL 注入漏洞会直接泄露应用系统数据

	SQL 注入漏洞可能导致系统数据被篡改 SQL 注入漏洞可能导致服务器沦陷 .....
漏洞原理	SQL 注入是由于应用程序未对用户输入进行合法校验，程序员在编写 SQL 语句相关代码时，未采用安全编码，导致攻击者可以将恶意 SQL 语句拼接至应用的 SQL 语句中，从而在数据库中执行，以此达到攻击目的。
使用安全编码	程序员可以通过使用安全编码来修复 SQL 注入漏洞。如下为修复场景： 当应用系统采用字节拼接 SQL 查询字符的方式会产生 SQL 注入漏洞，主要原因是用户进行的查询参数包含了恶意的 SQL 语句，而应用程序也没有控制输入去限制非法输入，那么就会直接形成恶意语句拼接到应用程序 SQL 语句中在数据库执行的情况，从而产生 SQL 注入漏洞。因此程序员可以使用参数化查询（Prepared Statements）或存储过程编码来替代直接拼接 SQL 查询字符串。使用参数化查询或存储过程来执行数据库查询操作。这样可以将用户输入的数据作为参数传递给查询，而不是直接拼接到查询字符串中，从而修复 SQL 注入漏洞。

3) 使用安全设备

企业可以通过合理使用安全设备来解决大部分安全漏洞的威胁，比如应用系统存在一些应用漏洞可以通过部署 Web 安全应用防火墙来防御相应的攻击。

以下是一些常见的安全设备及其作用：

a) 防火墙（Firewall）：是一种位于内部网络与外部网络之间的网络安全系统，可以将内部网络与内部网络隔离。它一般用于监控和控制网络流量，可以有效阻止未经授权的访问和恶意攻击。

b) 入侵检测和防御系统（IDS/IPS）：入侵检测系统（IDS）是一种用于监控网络行为的安全系统，一旦发现异常就会发出告警。而入侵防御系统（IPS）

在基于入侵检测系统的基础上，带有了阻断功能，也就是说当入侵检测系统检测攻击行为后，只要部署合理是会进行拦截并生成相应日志。

c) Web 应用程序防火墙（WAF）：是一种网络安全设备，主要用于保护 Web 应用程序免受各种 Web 攻击，如 SQL 注入、跨站脚本、CSRF 等

d) RASP（运行时自我保护）：RASP 技术通常内置在一个应用程序或应用程序运行时缓解中，能够控制应用程序的执行。当应用程序运行时，RASP 可以通过分析应用程序的行为和该行为的上下文，保护其不受恶意输入或行为的影响。 RASP 通过使应用程序持续检测自身的行为，可以立即识别和缓解攻击，且无需人工干预。RASP 与 WAF 相比，成本更低，而且可以在应用底层解决漏洞。但是 RASP 可能会给应用服务器带来性能消耗。

e) 安全网关（Secure Gateway）：安全网关是一种多功能装置，它同时具备了网络防火墙功能。网络入侵检测功能以及防病毒功能等等。安全网关可以提供安全的访问和流量控制，同时提供身份验证、访问控制和加密等功能，保护系统免受恶意攻击。

以上只是众多安全设备中的冰山一角，不同的安全设备具备的安全能力也不一样。企业可以通过部署不同的设备来解决不同安全风险，这些都需要结合企业的实际情况进行部署，比如企业存在应用漏洞，但是无法短时间修复，又未购买应用防火墙，这时我们可以通过部署软 WAF 来起到应用层的防护效果，虽然软 WAF 可能存在一定的防护效果差等缺点，但是依然可以拦截大部分攻击。类似这种的情况很多，再比如企业经常遭受恶意流量攻击，那就可以部署入侵防御系统做流量检测和阻断来解决相应的威胁。

4) 安全加固配置

安全加固配置是指在系统、应用程序、网络设备登录各个层面上，通过采取一系列的配置措施来增强安全性。安全加固配置有时可以是一种漏洞修复方法，也可以作为漏洞的缓解方式。针对安全漏洞，需要安全人员给出对应的漏洞修复方式，并全面分析系统的产生原因、触发原理，然后进行针对性配置来修复漏洞或临时缓解漏洞威胁。

以下是一些常见的安全加固配置：

a) 强密码策略：通过实施强密码策略，要求用户使用复杂且不易猜测的密码，加强身份验证和防止密码猜测攻击。

b) 访问控制和权限管理：限制用户和系统的访问权限，确保只有授权的用户可以访问敏感信息和执行受限操作，可有效解决未授权访问与越权相关漏洞。

c) 安全更新和补丁管理：定期更新和升级系统、应用程序和设备，确保应用最新的补丁和安全更新，修复已知的漏洞。

d) 网络安全配置：配置网络设备、防火墙和路由器等，限制网络流量、过滤恶意流量，并实施网络隔离和分段，减少攻击面。

e) 安全审计和日志管理：启用安全审计和日志记录功能，监控和分析系统日志，及时发现和响应异常行为和潜在的安全事件。

f) 禁用不必要的服务和端口：关闭和禁用不必要的网络服务和端口，可减少攻击者利用的机会。

g) 加密通信：使用安全的通信协议（如 TLS/SSL），对敏感数据进行加密传输，防止数据在传输过程中被窃取或篡改。

h) 身份验证和访问管理：实施双因素身份验证、单点登录（SSO）等安全机制，确保只有授权的用户可以访问系统和数据，可有效解决登录认证、授权访问类漏洞。

i) 权限最小化：将最小权限原则应用于用户和服务的配置。限制用户和服务的访问权限，只赋予其执行任务所需的最低权限，以减少潜在的攻击面。本条适用于由于用户或服务权限过高导致的安全漏洞修复。

安全加固配置在实际应用中并非是简简单单的加一条配置项就能起到防御效果，而且要对漏洞进行深入分析，确定漏洞产生原因，漏洞触发条件，然后再通过修改配置，将触发条件改为不满足，这样才算是真正的通过安全加固来修复或缓解漏洞。

例如，log4j2 在 2021 年底爆出编号为 CVE-2021-44228 的漏洞，Apache Log4j2 是一个基于 Java 的日志记录工具。CVE-2021-44228 漏洞形成的主要原因

是 log4j 的接收器对于不可靠来源的输入没有进行过滤，攻击者可以利用此特性构造特殊的请求数据包，以此来触发远程代码执行。按照正常思路来说这类漏洞直接进行 log4j2 组件升级即可修复漏洞，但是有很多业务担心组件升级带来不兼容等风险，于是组件升级对于部分用户来说就不是最优选择，而这时也可以采用安全配置来解决该漏洞。CVE-2021-44228 漏洞的触发点 JndiManager.lookup，因此也可以通过以下配置起到防护作用：

①设置参数：log4j2.formatMsgNoLookups=True

②修改 jvm 参数：-Dlog4j2.formatMsgNoLookups=true

③系统环境变量：FORMAT_MESSAGES_PATTERN_DISABLE_LOOKUPS 设置为 true

5）安全设备配置

安全设备在未进行精细化的安全运营的情况下，是无法最大化的发挥这些设备的价值的。在实际工作中可以通过对安全设备的精细化运营来解决一些不太好解决的安全风险或威胁。如企业可能存在一些不具备控制权限的资产（第三方资产），如果该类资产存在安全漏洞，我们就不具备直接修复漏洞的主动权，这时就可以通过增加一些安全设备的配置来降低该类漏洞的威胁。

下面介绍一些通过配置安全设备来加强企业整体安全性或调优的内容：

a） 内部网络安全：通过配置防火墙实现网络隔离，可以将内部网络划分为不同的安全区域或子网，限制网络流量和访问，防止内部网络受到外部威胁的影响。当内网出现类似网络病毒感染等攻击时，该手段可将攻击范围缩小至可控范围，可避免病毒大面积扩散。

b） 恶意流量和攻击阻止：部署 IPS 入侵检测系统，启用攻击阻断功能，可以监控和过滤网络流量，阻止恶意流量和攻击，能减少网络威胁对系统的影响。

c） 未经授权访问阻止：通过配置访问控制列表（ACL）和防火墙规则，可以限制网络访问权限，阻止未经授权的用户或系统访问敏感数据和资源。

d） 弱点利用防护：Web 应用防火墙可以阻止常见的攻击和漏洞利用，如 SQL 注入、跨站脚本攻击（XSS）、跨站请求伪造（CSRF）等，保护应用程序

和系统免受攻击，我们安全工程师可以通过配置 WAF 规则，不断降低 WAF 的误报率，使 WAF 的价值不断提升，也可以在遇到一些 0day 时，在暂无其它解决手段的情况下，可以通过新增 WAF 规则来检测拦截相应的攻击。

e） 安全审计和日志记录：防火墙可以记录网络流量、事件和安全日志，提供审计和调查所需的信息，帮助发现和响应安全事件，确认漏洞攻击源，攻击方式，可以此作为参考编写新的防火墙规则，完善防火墙防护功能。

每个企业都会在互联网出口部署防火墙，但是部署了防火墙并不意味着防火墙后面的网络环境就是安全的，因为很有可能出现一些敏感端口向公网开放的情况。

例如，企业在公网开放了 9200 端口，该端口存在 ES 的未授权访问漏洞，技术人员直接为 ES 配置鉴权时，会因为其它应用调用了 ES，直接影响业务。这时我们可以通过配置防护墙来限制 ES 的访问范围，以达到临时的缓解作用，这样就可以有充足的时间进行整改工作。

企业中的安全漏洞、威胁是非常复杂的，部署安全设备是解决这些漏洞、威胁非常重要的一种手段，但仅仅拥有安全设备是不够的，还需要专业的技术团队来进行运营。

6）认证与授权

认证与授权机制是保护系统免受恶意用户和攻击的重要手段，但仍需要与其他安全措施（如输入验证、安全编码等）相互配合，以构建综合的安全防护措施。可以通过设置合理的认证与授权机制帮助企业解决许多安全漏洞和减少潜在的安全威胁，包括但不限于以下内容：

a） 减少弱密码和密码猜测攻击：可以通过实施强密码策略和合理认证机制（如多因素认证），如在新增用户与修复密码处增加强口令检测机制后，用户输入密码都必须满足强口令策略，这样直接减少了弱口令的产生。我们在登录入口设置双因素认证，不但验证用户名和密码还验证一次性密码（如 OTP），在通过的情况下才允许登录，这样即使系统存在弱口令用户，也能直接防御弱口令用户登录系统。

b) 防止未经授权访问：我们可以通过合理的认证与授权机制，来精细化管理应用的每一个服务，保证只有被授权的用户才能获得特定资源的访问权限，可有效防止未经授权的访问和数据泄露。

c) 防止水平/垂直权限提升：我们可以通过合理的认证与授权机制，在每次权限检测时都不只是单单通过一个 ID、一个用户名就通过验证，而应该严格检测当前用户生成会话信息（sessionis、token 等）。避免恶意用户通过修改一个 id 或用户名就可以进行越权攻击，获取比其应有权限更高的访问权限，确保用户只能访问其所需的资源和操作，防止特权滥用。

d) 防止跨站点请求伪造（CSRF）：我们在设置认证授权时为每个表单中都增加一个 CSRF token，以此来确保表单提交的请求数据来自合法用户，防止恶意请求的执行。

通过上面的描述可以看到认证与授权非常的重要，接下来通过基于角色的访问控制模型解决垂直越权漏洞来深入了解认证与授权的有效性。

基于角色的访问控制模型是指角色与权限关联，用户通过适当角色的成员，得到角色的权限。根据角色授权，可以控制减少授权的数量，配置复杂度低，灵活性好，可以组成结构层次。这种配置可以简化权限管理，并确保用户只能访问其所需的资源和操作。

垂直越权漏洞是指低权限用户可以访问或操作只有高权限用户才能访问或操作的内容。比如查看系统用户列表这个功能一般只有系统管理员才可以查询，但是存在垂直越权的系统，普通用户在获取到该查询接口后可以直接通过普通权限获取到系统用户列表。垂直越权漏洞形成的主要原因是程序员在进行权限控制时，只在菜单加载时检测了用户的权限，并没有在接口处验证用户的权限。

通过在系统中实现基于角色的访问控制模型，这样为管理员设置相应的管理角色，普通用户设置普用户角色，这样为不同的角色配置不同的权限，然后用户根据自己的角色获取相对应的权限，用完普通用户无法获取到管理员角色的权限，也就没法再去越权查看管理员才能查看的内容了，因此也就完成了垂直越权漏洞的修复工作。

7）程序更新

程序更新是解决安全漏洞的关键技术之一,可以通过更新程序(组件、软件、系统、补丁等）来解决已知安全漏洞和潜在的威胁。但是程序更新并不意味者可以解决所有的安全漏洞，但它是保持系统和软件安全的重要手段之一。

程序更新可以解决的问题包括但不限于以下内容：

a) 已知漏洞修复：程序更新通常包含已知漏洞的修复补丁，这些漏洞可能被攻击者利用来入侵系统、执行恶意代码或者获取敏感信息，更新可以应用这些修复补丁，增强系统的安全性。

b) 弱点和错误修复:更新还可以修复软件中的弱点和错误,包括编码错误、配置错误等，这些弱点可能被攻击者利用来绕过安全措施或者执行未经授权的操作。

c) 安全功能改进：程序更新还可以引入新的安全功能和机制，以应对新的威胁和攻击方法,这些功能改进可以增加系统的抵御能力,提高安全性。

d) 恶意软件防护：程序更新还可以更新防病毒软件、反恶意软件工具等，以识别和防护新出现的恶意软件和病毒，保护系统免受恶意软件的感染和攻击。

上文在安全加固配置部分提到可以通过安全配置来修复 2021 年 log4j2 产生的漏洞 CVE-2021-22248，另外的修复该漏洞的方式便是通过修改配置文件（入 pom.xml），将 log4j2 的版本改为安全版本即可解决该漏洞。

每一个漏洞是有多种修复方式的，企业是需要结合实际情况去制定合适的漏洞修复方式的，“随大众”使用一些通用的漏洞修复方式去修复漏洞可能造成资源浪费，人力、时间投入过大的情况。

五、 持续漏洞管理

企业在进行漏洞修复的同时也需要做好持续漏洞管理工作,漏洞修复工作只是解决了目前发现的安全漏洞,而新发现的漏洞以及潜在的风险都可能给企业带

来很大损失。

持续漏洞管理需要综合考虑多个因素，包括技术、流程和人员方面，它是保持系统安全性的一种能力，应与其他安全措施和最佳实践相结合，以构建全面的安全防护体系。

以下是持续漏洞管理需要做的工作：

1) 漏洞扫描和评估：定期进行漏洞扫描和评估，通过渗透测试、代码审计、开源风险扫描、漏洞扫描等手段，识别系统和应用程序中的漏洞和安全弱点。

2) 漏洞修复和补丁管理：根据漏洞扫描结果，及时修复和应用相关的安全补丁，确保系统和应用程序免受已知漏洞的影响。

3) 漏洞跟踪和管理：进行漏洞跟踪，提供最新漏洞信息，包括漏洞的描述、风险评估、修复进度等信息。

4) 漏洞优先级和风险评估：依据上文提到的安全漏洞修复策略的来确定漏洞修复的优先级。

5) 漏洞修复计划和策略：依据上文提到的安全漏洞修复策略来制定漏洞修复计划和策略，确保漏洞得到及时修复，并提供明确的责任分工和时间表。

6) 安全补丁管理流程：建立有效的安全补丁管理流程，包括补丁的获取、测试、部署和验证，以确保补丁的正确应用和系统的稳定性。

7) 漏洞验证和渗透测试：进行漏洞验证和渗透测试，通过模拟真实攻击场景来验证修复的漏洞，并发现新的潜在漏洞。

8) 安全意识培训和教育：定期进行安全意识培训和教育，帮助用户和开发人员识别和应对漏洞，促进整体安全文化的建立。

9) 持续改进和更新：持续改进漏洞管理流程，根据新的安全威胁和最佳实践，更新和改进漏洞管理策略和工作流程。

六、 总结

企业安全漏洞修复是保护企业数据、系统和声誉的关键步骤。修复安全漏洞可以减少数据泄露、系统瘫痪和恶意攻击的风险，降低企业面临的法律和合规性

问题。它不仅可以保护企业的利益和客户信任,还可以防止声誉损害和财务损失。通过综合的修复策略和措施，包括但不限于补丁管理、配置调整、程序更新、代码加固。企业可以有效降低潜在的安全风险，并建立一个安全可信赖的环境。

通过修复企业安全漏洞，企业可以实现以下重要目标：

1) 数据保护：修复安全漏洞可以防止敏感数据的泄露和未经授权的访问。这样可以确保客户、员工和业务数据的安全性和机密性，避免潜在的法律和合规风险。
2) 系统稳定性：修复安全漏洞有助于提高系统的稳定性和可靠性。通过修复潜在的漏洞，可以减少系统崩溃、拒绝服务和其他安全事件的风险，确保业务的持续运行。
3) 防止业务中断：修复安全漏洞可以防止恶意攻击导致的业务中断。安全漏洞可能导致系统故障、服务不可用或数据丢失，影响业务运营。通过及时修复漏洞，企业可以确保业务的连续性和可靠性。
4) 品牌声誉保护：修复安全漏洞有助于保护企业的品牌声誉。通过修复漏洞，企业能够展现对客户数据和隐私的关注，并树立信任和可信度，增强品牌形象。
5) 避免金融损失：修复安全漏洞可以帮助企业避免潜在的金融损失。安全漏洞可能导致恶意活动，如数据盗窃、支付欺诈等，造成财务损失。通过修复漏洞，企业可以降低金融风险，保护财务利益。
6) 维护客户信任：修复安全漏洞有助于维护客户的信任。客户对于其数据的安全和隐私非常关注，如果企业能够积极修复漏洞并保护客户数据，将增强客户对企业的信心，并增加客户忠诚度。
7) 遵守行业标准：修复安全漏洞有助于企业遵守行业标准和最佳实践。许多行业都有特定的安全要求和标准，通过修复漏洞，企业能够符合这些要求，提高行业声誉和竞争力。
8) 合规性和法规要求：修复安全漏洞可以使企业符合法规和合规要求。许多法规和标准要求企业采取适当的安全措施来保护客户数据，修复漏洞是符合这些要求的重要步骤。