

# 安全架构要参

构建企业适用的安全架构

*Kunpeng Zhao*

*Senior Security Engineer | 2021/12*

# 目录

声明 .....	ERROR! BOOKMARK NOT DEFINED.
序言 .....	4
第一部分——闲话安全工作 .....	5
1. 一些基本观点 .....	5
2. 认识业务 .....	6
3. 认识组织架构 .....	7
4. 阴与阳 .....	8
第二部分——浅谈安全架构 .....	10
5. 认识架构 .....	10
6. 认识企业安全 .....	14
7. 安全架构基础 .....	22
8. 解决方案 .....	24
第三部分——具体实施内容 .....	26
9. 持续服务 .....	26
10. 成为安全架构师 .....	27
附录 <sup>1</sup> .....	28
A. 推荐阅读 .....	28

---

<sup>1</sup> 名词术语应自行查阅，不做附录。

# 声明

本书受限于作者自身水平，依然会有一些问题存在，需要读者客观看待。

本书旨在帮助读者构建自己的安全架构理念，结合所在企业实际情况去设计相应的治理方案。

本书的读者定位在甲方的安全架构师以及具有几年工作经验的安全工程师上，见贤思齐，见不贤而内自省。

本书不会专注在具体工具的使用之上，或者是某个具体技术细节之上，或者是如何绘制架构图等。

本书对于有些英文并未意译，建议读者记住此类单词原意。

本书在写作之初增加些许引用及参考，读者阅读需对参考文档细致学习。

本书力求言简意核，全文共分三个部分：第一部分（一到四章）介绍了安全工作的一些观点和经验，第二部分（五到八章）讨论了架构的定义、安全的定义和具体工作以及如何制定安全解决方案，第三部分则是具体介绍了如何实施以及持续交付。

# 序言

近年来市面上中文类安全书籍大有进发之势，纵观所读，大多着眼于工具使用亦或某一领域的技术介绍。谈及安全架构必然长篇大论，洋洋洒洒不尽其数。质量上乘者却多以译本<sup>2</sup>为主，作品又极少。笔者原亦想增砖添瓦，奈何才疏学浅。因此虽有时成书之意，却无动笔之宜。疫情以来，无奈居家办公，得闲整理之前所学，动笔之意渐强，便顺其自然打算写一本中文安全书籍。

动笔之初，原想借此前手稿写一本企业安全类相关书籍。后审阅再三，仍觉个人水平不足。故缩小范围，仅就安全架构相关成文。时值晚秋，大体部分已经写完，回顾来看却仅有万余字，说成书倒不如成文合适。心中犹豫，能以书论否？后又雕琢，发现文字愈少，亦愈发忐忑。隧请教一位编辑，“问诊”之后建议我将骨干铺开，细节引申，填充文章之血肉，使其饱满。感谢之余确有受益。后再次审阅全文，愈进行增补删改，忽觉背离本意。原意在梳理安全架构之方方面面，启发读者自行组装框架，有所参考，但又不拘束于此。自省二三，我何以看重自己写的是个书，还是个册子呢？正所谓破山中贼易，破心中贼难。又过了几天，笔者再次阅读手稿，执念渐淡。但求行文有效，倒也不再拘泥字数与否。细想之，不过一本拙作而已，抛砖引玉罢了。

今年以来天灾人祸，诸事繁杂<sup>3</sup>。心中闷闷，却也最终完成写作。笔者深知自身资历平平，成文之后难免有所疏漏亦或观点错误。亦不知此书有人阅读与否，如有不吝赐教<sup>4</sup>者自当十分感谢。

---

<sup>2</sup> 倒不是说外来的和尚会念经，只是他们已经有足够土壤成长，足够的时间去沉淀。

<sup>3</sup> 感谢家人与亲友的鼓励与帮助。

<sup>4</sup> 笔者邮箱：mour@iami.xyz

# 第一章：一些基本观点

本章先去尝试和读者确立起相同的或者类似的观点，虽然说推荐辩证思考，但如果基本观点不相同的话，那很显然说明这套方法可能对您起不到多少帮助。这里的观点虽然大都与技术无关，但需要每个安全架构师都能时常记住。

三人行，必有我师，择其善者而从之，其不善者而改之。

——《论语》

## 业务形态决定 IT 设施

业务形态决定了 IT 的形态<sup>5</sup>，IT 建设需要围绕业务进行，更遑论安全。

## 没有绝对安全的系统

没有绝对安全的系统，任何系统都存在被攻破的可能性。无论是通过技术手段还是社会工程学等，总有办法被攻破。

## 知道你的 Scope<sup>6</sup>

知道团队、个人的职责范围，权力范围。如果不知道，务必先明确之<sup>7</sup>。

## 关注需要的资源

当知道有多少预算时才知道能搭建多大的团队、采用什么级别的产品、提供什么样的服务、需要多少资源、多少时间、是否能够得到上级的充分支持，合作团队的配合等。

## 关注生态

业务及业务周边，同样，安全能力建设中的合作部门，安全产品厂商，白帽子阵营等等。

---

<sup>5</sup> 曾有一部分讨论是监管和业务形态的关系，可以确定的是业务所在的赛道明确了企业可能受到的监管，这是赛道的规则。也许有时还没有规则，但最终肯定是要有的。

<sup>6</sup> 未把 Scope 翻译成范围。

<sup>7</sup> 安全建设不是一个团队的事情，是整个企业所有员工需要共同参与。

# 第二章：认识业务

本章简单介绍了认识业务的一些简单方式，只有通过对业务的了解才能完成对业务的安全建设防护。

知彼知己，百战不殆。<sup>8</sup>

——《孙子·谋攻篇》

## 了解企业所在行业的基础知识

充分了解企业所在行业的相关知识、最新的法规条例、如何演变的业务形态、关键的系统以及通讯协议、具体的业务流程以及制度等；同时关注行业领域内 IT 的发展变化、IT 所占业务盈利的百分比、IT 中安全预算的占比以及关注同行业 CIO, CTO 对技术趋势的倾向性。了解行业基础，了解业务的核心价值以及发展趋势。业务核心是需要安全架构主动关注的，长期关注的部分。假设一个企业的优势在于跨境支付，那么围绕跨境支付的所有业务以及对应的系统就是核心价值所在。

## 关注企业的财报

关注企业的盈利能力，公开的财务数据能够知道企业内部是不是足够透明的，良性竞争的企业内部向来是不吝去开诚布公的讨论这些指标。员工经常能在 Dashboard 看到有多少笔 Transaction（交易），Volume（数额）。无论是公开的还是仅限内部的财务数据，只有在足够盈利或者有着充足储备的情况下，才能去建设安全团队以及完善安全体系建设。

## 了解你的客户

了解业务架构、应用架构、数据架构、技术架构等；了解业务的具体用户；了解用户的需求，结合现有的资源为客户提供服务。对于安全架构师来说，有时候并不是直接面临用户，而是企业内部的团队，客户不是平等<sup>9</sup>的，因为每个客户都是差异化的。

## 了解你的行业

了解安全行业的现状，以及各厂商的优劣，包含技术研究、售后支持、性价比等；

---

<sup>8</sup> 如果攻击面都不清楚，何谈防御呢？

<sup>9</sup> 指不具备相同的背景知识，例如不能奢求财务人员具备研发人员的背景知识，以及不奢求研发人员具备合规的知识。提供安全服务以相同的心态为不同水平的用户提供不同的服务。

# 第三章：认识组织架构

本章通过介绍常见的一些组织架构形式，来帮助读者认识协作的过程。

若网在纲，有条而不紊<sup>10</sup>。

——《尚书·盘庚上》

## 职能部门<sup>11</sup>

产品部门（设计）、技术部门（包含研发、运维、运营、安全、风控等）、法务部门、财务部门、市场部门、公关部门、行政部门（人事、党政）等。

## 协作方式

从职能层面来讲常见的方式为自顶向下<sup>12</sup>，即某委员会到某具体委员会到某具体工作组；例如：1. 企业数据委员会 → 2. 企业数据管理委员会 → 3. 企业数据管理工作小组；分别负责：

1. 定义原则战略政策以及做出决策、提供冲突解决；
2. 监督数据管理工作组、解决升级问题、报告关键结果等；
3. 对数据进行定义分层生命周期管理、执行变更、评估质量等；

类似的还有企业架构委员会→应用架构委员会→架构评审工作小组<sup>13</sup>等；除此之外对于集团性质的企业还存在虚实线汇报，本地化实线汇报以及集团的虚线汇报。推进某些项目时由工作小组牵头，同级之间分发工作到具体团队。在某些大型项目上往往还需要指定具体的项目经理跟踪整个进度，在日常工作中遇到资源不足时向上请求支持。当然现在还有一种能力中心三驾马车(BP、SSC、COE)的运营模式<sup>14</sup>，

## 安全团队<sup>15</sup>

当资源充足的时候，安全团队自身就是具备架构师，工程师，产品经理，项目经理等。在集团化企业中，会设置不同的安全防线分别用于运营及管理、内控、内审。不同的部门之间互相制约，共同建设安全体系。例如内审团队可以通过定期审计，促使团队优化 Policy 以及 SOP。值得注意的是，由于不同行业的关注点不尽相同，因此安全团队的组成也并不相同<sup>16</sup>。

## 企业文化<sup>17</sup>

小到团队氛围大到企业文化，想必大多数读者应该是经历过不少人情世故。喜乐悲愁，自是别有一番滋味。广义正确的说法是每个人都应该始终保持身体健康，对客户友好，对同事宽容，使用创新技术。远离 996，远离 PUA，良性竞争，避免无用消耗。

<sup>10</sup> 事实上，企业的组织架构不用追求办起事情了 100% 有条不紊，能把常见的事情做到已经不易。

<sup>11</sup> 形式上的划分一般有金字塔型（职能制、事业部制、矩阵制等）和扁平化。

<sup>12</sup> 架构设计强依赖自顶向下。自底向上的工作方式在大企业内可能性微乎其微。

<sup>13</sup> 例如由安全架构师、研发架构师、数据架构师、运维架构师组成。

<sup>14</sup> 三驾马车与信息安全 BP <https://zhuanlan.zhihu.com/p/41363175>

<sup>15</sup> 详参 <https://iami.xyz/MY-Enterprise-Cyber-Security-Architecture/> 第三节。

<sup>16</sup> 更常见的是团队职责和个人承担的职责不匹配。

<sup>17</sup> 可阅读《基业长青》加深对企业文化作用的理解。

# 第四章：阴与阳

本章通过简单介绍阴阳的特性，希望籍此能够消解工作中的一些情绪，从而改变自我的认知以及帮助自己与团队进步。其后针对一些争议性问题给出了浅薄参考。

道可道，非常道；名可名，非常名。

——《道德经》<sup>18</sup>

## 对立制约

相互对抗，互相制约

以攻击与防守为例；有攻击者就会有防守者，有防守方式就会有破解的欲望。不过虽说攻击与防守互相制约，但不得不承认黑产永远是走在最前面的。无他，利益驱动。

## 互藏互化

互藏互育，相互转化

以甲方与乙方为例；攻击者可以服务厂商，也可以去甲方内部。防守者即可以服务企业，也可以把经验沉淀出产品。防守者获得厂商的服务同时又向企业内部提供服务。

## 互源互用

相互依存，交错运用

以企业和监管部门为例；企业和个人一同组成行业协会，协会帮助监管部门完善法律法规，征求专业人士的意见。反之最新的标准又帮助企业建设自身，帮助行业更加规范。

## 消长平衡

此消彼长，此长彼消

以价值与风险为例；随着价值的逐步增加，面临的风险就会变大。攻击者大多都不会浪费资源对没有价值的目标下手。诈骗份子也不会去将流浪汉作为犯罪目标。企业代表的利益越来越多，就会面临国家的监管。小到消防审查，大到反垄断法。当带来的价值和未知的风险处相对平衡才是理想的。

## 思考一些问题

这里面可以讨论的包括部门、技术、员工（信心）、资源、监管、企业、行业、国内、国际、商誉、舆论、攻击和防御等；而这许多事情都需要辩证看待，某脉上过去两年最常看到的讨论就是入职某企业在集团内是要低人一等的，开启劝退。简单的思考应该要先看整体环境，再看部门。确定是否整体向着好的方向发展，有没有影响你本来的目的，薪水充足与否，小团队范

<sup>18</sup> <https://www.zhihu.com/question/20175885>

围怎么样，技术有无成长等。此处列举了一些问题，可以在下面的空白处，留下你自己的思考。

## 1. 如何看待自身成长

引申：你真正需要的是什么？

➤

## 2. 如何看待新技术

引申：哪些需要紧跟行业最佳实践？如何分辨哪些是当前需要的？

➤

## 3. 如何看待遗留系统

引申：遗留系统之所以遗留是具备历史因素的

➤

## 4. 如何看待团队内以及团队间的协作

➤

## 5. 如何解决“研发看起来一点也不懂安全”

引申：也许不止研发是吗？

➤

## 6. 是选择脚本还是工程化代码

引申：是否要选择临时性的解决方案？什么时候会选择？

➤

## 7. 如何看待行业大环境

➤

## 8. 如何看待“管理者不够聪明”

➤

## 9. 如何看待寻找新机会

➤

# 第五章：认识架构

本章先是针对架构定义、目标、规范、质量进行介绍，其后分别针对业务架构、应用架构、数据架构、技术架构进行介绍。

一阴一阳之谓道，继之者善也，成之者性也。

——《易经·系词上》

## 定义

*The art or science of building* 一种关于构建的科学或者艺术

将构建目标的各种依赖<sup>19</sup>（虚）拆解出来并将各对应组件（实）通过某种方式运转起来以符合目标需求的一种能力。<sup>20</sup>

## 目标

建立安全、可靠、高性能且符合成本控制的系统架构。

## 规范

需要具备使用指南、参考架构、以及经过审查的工具和代码，同时需要满足可操作性原则。

## 质量<sup>21</sup>

在评估架构质量时需要考虑的有以下范畴。通用的讲需要满足三个方面：

- 安全（Security）
- 合规（Legal & Regulatory）
- 灾难恢复（DR）

针对外提供的服务需要满足可用性（Availability）、性能（Performance）、容量（Capacity）、隔离（Isolation）、可视化（Visibility）；

针对内部的或者说本地化的需要满足

- 成本效率（Cost Efficiency）
- 可扩展性（Extensibility）
- 可操作性（Operability）
- 可维护性（Maintainability）
- 可伸缩性（Scalability）

### Architecture Quality

- Scalability
  - Capacity
  - Scalability
  - Visibility
  - Performance
  - Efficiency
- Stability
  - Availability
  - Operability
  - Isolation
  - Maintainability
  - DR
- Speed
  - Extensibility
  - Testing Strategy
- Simplicity
  - Cost Effectiveness
  - Reduced Technical debt
- Security
  - Compliance
  - Security

图 5-1

<sup>19</sup> 依赖包含了逻辑上的所需而后映射到具体实际产品、组件，通过各种技术、流程将其运行起来以及持续运营并提供服务。

<sup>20</sup> 本定义是结合作者自身理解得出，仅供参考。

<sup>21</sup> 并非所有质量均需满足，读者需结合实际场景判断其需要支持的特性，随后采取对应技术支持该特性。例如保持 HA，有负载均衡，有分级处理，有超时，有服务降级等。

# 业务架构

Know your Business

业务架构<sup>22</sup>是代表整体的、多维的业务视图<sup>23</sup>。包含能力、端到端的价值交付、信息和组织架构、以及战略、产品、政策、计划和利益相关者之间的关系，并对业务流程功能进行分解。主要目标和原则为以下几点：

- 遵守法律：企业政策是遵守法律、政策和法规。企业信息管理流程符合所有相关法律、政策和法规。企业必须注意遵守有关数据收集、保留和管理的法律、法规和外部政策。
- 保持业务连续性：必须评估应用程序的关键性和对企业任务的影响，以确定需要何种程度的连续性以及需要何种相应的恢复计划。
- 为企业带来最大利益：从企业范围的角度做出的决策比从任何特定组织角度做出的决策具有更大的长期价值。
- 确保 IT 和业务相适应<sup>24</sup>：IT 组织负责拥有和实施 IT 流程和基础架构，使解决方案能够满足用户定义的功能、服务级别、成本和交付时间要求。

下图阐释了个业务架构中各元素之间的具体依赖。

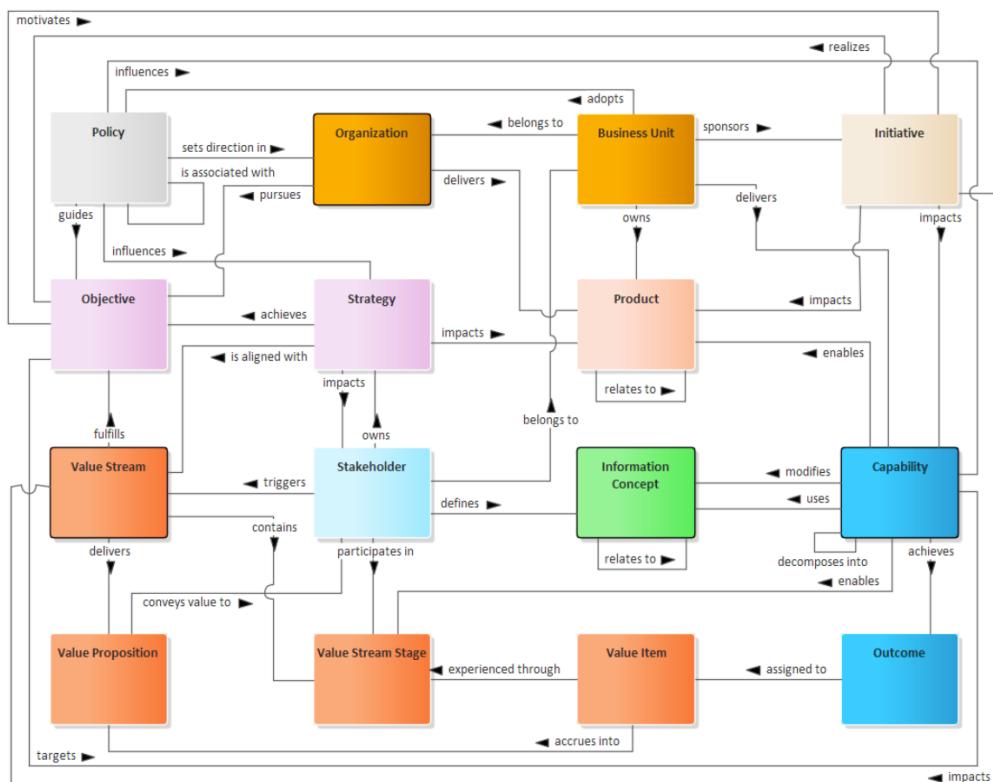


图 5-2

<sup>22</sup> 另和业务架构一同提起的还有产品架构，属于是讲具体业务功能流程拆分后的产物。不在此赘述。

<sup>23</sup> 引用自

[https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/public\\_resources/Business\\_Architecture\\_Metamodel.pdf](https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/public_resources/Business_Architecture_Metamodel.pdf)

<sup>24</sup> 这是 CIO 的主要职责之一。

## 数据架构

*Know your Data*

通过组织架构，技术管理，流程控制，策略标准等更新迭代去实现数据治理。企业数据策略通过驱动数据治理以实现商业价值。针对数据管理，详参以下各个方面。

### 数据治理

- 组织架构及运营模型
- 策略及流程
- 数据域模型及所有者
- 数据问题管理
- 数据变更管理

### 数据质量

- 数据分析
- 业务规则和阈值
- 数据清洗
- 数据补救
- 数据质量报告

### 元数据

- 业务分类
- 数据字典
- 元数据管理维护
- 元数据访问

### 数据风险管理

- 风险管理

### 数据保护

- 数据安全
- 数据隐私<sup>25</sup>

### 主从数据

- 标准及聚合
- 业务和数据规则
- 数据中心和常用服务
- 主从数据持久化
- 主从数据访问

### 数据运营

- 数据生命周期管理<sup>26</sup>
- 数据供应及源认证
- 数据转移和持久化
- SLA 管理
- 数据认证

### 平台及架构

- 数据模型
- 数据管理平台
- 数据整合
- 数据架构

## 应用架构

*Know your Services*

此处主要针对单一系统<sup>27</sup>的应用架构进行介绍，在单一系统的应用架构中，实现了技术层面的在类和代码的层级上有：

- SRP（单一职责原则）
- OCP（开闭原则）
- LSP（里氏替换原则）
- ISP（接口隔离原则）
- DIP（依赖反转原则）

<sup>25</sup> Data privacy is not about data, but about person.

<sup>26</sup> 主要包括收集、存储、使用、分享、销毁；更多内容可参考 DSMM 模型，但实际治理不能仅以生命周期为主。

<sup>27</sup> 企业级应用架构实现了将业务流程映射到具体 IT 设施的过程。

在组件的层级上有：

- REP (复用、发布等同原则)
- CCP (共同闭包原则)
- CRP (共同复用原则)

处理组件依赖问题的三原则：

- 无依赖环原则
- 稳定依赖原则
- 稳定抽象原则

## 技术架构

*Know your Technology*

技术架构一般是包含了整个技术范畴，在蓝图级别将整体的业务架构映射到具体的IT技术设施上。通常由架构师委员会<sup>28</sup>制定，包含了基础设施的技术栈（两种左右的编程语言，常用框架，数据库等），系统设计（Restful, SOA, SOAP, MicroService等），预研方向等。此处以金融支付类企业举例。

基础设施：基础设施，数据平台，开发平台，集团服务等；

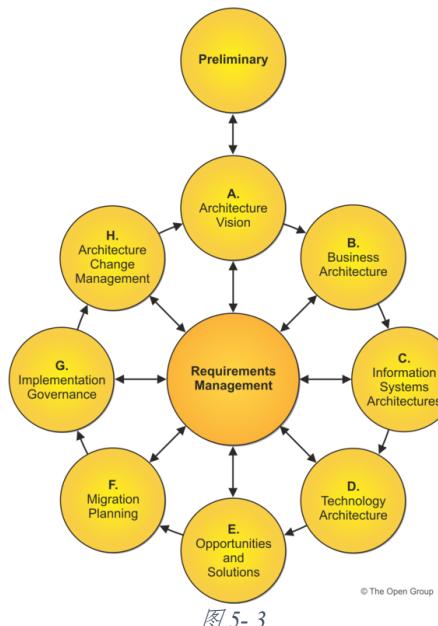
通用平台：卡，风险，支付，认证，合规，金融，客户服务等；

产品线：具体到各个产品线等；

客户：消费者，商户（小微，企业，合作伙伴），全球本地化等；

## 软件架构参考框架<sup>29</sup>

TOGAF<sup>30</sup>



<sup>28</sup> 参考第三章协作方式

<sup>29</sup> IT管理类框架、行业合规类框架，均不在此处阐述；Zachman亦不介绍了。

<sup>30</sup> 详细参考 Opengroup 官网 <https://pubs.opengroup.org/architecture/togaf9-doc/arch/pt1.html>

# 第六章：认识企业安全

本章先简单介绍安全及威胁的特性，其后针对安全治理进行介绍。包含治理范围，设计原则以及技术范畴、合规范畴、管理范畴、运营范畴等相关领域。通过安全治理为企业架构增加安全性，从而构建安全，可靠，高性能并满足成本控制的架构系统。

不谋万世者，不足谋一时；不谋全局者，不足谋一域。<sup>31</sup>

—— 陈澹然

## 定义

在各方面不同方式<sup>32</sup>为企业架构提供安全特性并通过各种机制确保流程<sup>33</sup>正常。

## 安全特性

- 机密性(Confidentiality): 确保数据在传输和存储时的机密性，避免未经授权的用户有意或无意地泄露数据。
- 完整性(Integrity) ): 确保数据在传输或存储的生命周期内，始终保持正确性和一致性。
- 可用性(Availability): 确保用户在需要通过信息系统进行操作时，数据和服务必须保持可用并满足需求。

除 CIA 之外还有 3A<sup>34</sup>，认证——识别信息用户的身分，并记录访问和使用信息；授权——根据实际需求给予实体授予适当的权限，一般采用最小权限；计帐——记录用户与系统间的交互数据。

## 威胁分类

- 欺骗
- 篡改
- 否认
- 信息暴露
- 拒绝服务
- 提升权限

## 安全治理

在不同领域通过技术、运营、管理的方式确保使企业架构具有安全特性<sup>35</sup>。

---

<sup>31</sup> 不乏许多企业并无任何安全意识，更别提谋一域安全，乃至全局安全。

<sup>32</sup> 包含不限于技术层面、流程层面、管理层面、运营层面等。

<sup>33</sup> 例如：数据流转，应用发布，业务上线等等

<sup>34</sup> Authentication, Authorization, Accounting。当然也有 5A 说法。

<sup>35</sup> 安保也算，虽然是行政或者物业管。

## 治理领域 (Scope<sup>36</sup>)

主要包含以下几点，另建议参考 AWS, Azure 等云厂商关注方向<sup>37</sup>。

- 合规 (Compliance)
- 数据保护 (Data Protection)
- 基础设施 (Infrastructure<sup>38</sup>)
- 身份认证和访问管理 (IAM)
- 应用和服务<sup>39</sup> (Application and Service)

## 驱动力<sup>40</sup>

基于事件的驱动往往意味着风险已经成为现实

1. 监管驱动  
执法必严，违法必究；
2. 事件驱动<sup>41</sup>  
吃一堑，长一智；
3. 价值驱动<sup>42</sup>  
未雨绸缪，上策；

## 安全设计原则

- 适应业务目标<sup>43</sup>
- 为攻击者设计
- 最低容忍<sup>44</sup>
- 最小化攻击面<sup>45</sup>
- 强身身份认证与权限管理
- 弹性设计<sup>46</sup>
- 安全左移<sup>47</sup>
- 纵深防御<sup>48</sup>
- 默认安全
- 安全内生<sup>49</sup>
- 让系统容易使用以及自动化<sup>50</sup>
- 平衡投资<sup>51</sup>

<sup>36</sup> 从技术视角来看，即包含基础安全，应用安全，数据安全等领域。

<sup>37</sup> 注意区分和所在企业的基础设施差距。

<sup>38</sup> 存储、计算、网络等。

<sup>39</sup> 需主动关注对应的供应链安全。

<sup>40</sup> 责任共担模型

<sup>41</sup> 亡羊补牢，未为晚矣。

<sup>42</sup> 资产就是价值集合，包含不限于商誉、业务、数据、IT 设施等。

<sup>43</sup> Information Security as the Enabler of Business.

<sup>44</sup> 具备基线 Baseline 以及基准 Bechmark。

<sup>45</sup> 梳理已知资产使其符合最低容忍能够有效降低风险；例最小化权限。

<sup>46</sup> Resilience design.

<sup>47</sup> 远至供应链，近至研发流程。即便概念已经是二十余年前提出，但至今仍未实际大范围落地。

<sup>48</sup> 采用分层、分级的方式。例从逻辑架构到实际系统架构、从概念架构映射到具体场景、从 High-Level Design 到 Low-Level 都应该考虑到安全特性。

<sup>49</sup> 可译成英文 Security by Origin。

<sup>50</sup> 低侵入性、低性能损耗、易于运营、易于持续维护等。

<sup>51</sup> 投资源（预算、人力、）到不同领域（技术、合规、运营等）的生命周期管理中。

## 合规范畴<sup>52</sup>

法务部门对接相关行政机关，转交给到对应合规部门<sup>53</sup>，随之根据行政法规，判断是否需要作出相应的调整去满足合规需求。例如是采购具备资质的乙方厂商的安全产品，或者使用指定的加密算法等。从分类上讲一般可划分为地域性的和行业性的：

### 1. 地域性合规

- 国际合规
- 本地化合规<sup>54</sup>
- 区域合规<sup>55</sup>

### 2. 行业性合规<sup>56</sup>

金融，医疗，保险，汽车等不同行业面临的行业合规标准也是不尽相同的。

## 技术范畴

技术是解决问题的第一生产力<sup>57</sup>

- 入侵检测和防御（Intrusion Detection and Protection）：例如入侵检测、文件完整性保护、威胁情报、态势感知等，以及对流量进行清洗（包含四层及七层流量清洗<sup>58</sup>）等；
- 集中化的日志管理（Centralized Log Collection）：例如 Splunk、ELK 等；
- 统一身份认证及权限管理（Identity and Access Management）：例如 AD、SSO、MFA、FIDO 等；
- 持续扫描及监控（Continuous Scanning and Monitoring）：例如针对文件、网络、主机、存储、应用、代码等资产<sup>59</sup>进行持续性的扫描及监控；
- 根信任（Root of Trust）：例如根密钥，随机数生成器，硬件加密模块，金杯体系；
- 公钥基础设施（Public-Key Infrastructure）
- 数据防泄漏（Data Leak Protection）：例如针对邮件、流量、文档；
- 安全开发套件（Security SDK）：例如加密套件、过滤器等；
- 运行时防护（Runtime Protection）：例如容器层级、应用层级；
- 创新技术的探索性应用：例如 Web3.0、AI、Blockchain、创新沙盒等；

## 运营范畴<sup>60</sup>

持续性维护及优化需要实现的安全目标

在持续的运营过程中交付安全能力并发现新的问题，以下尝试通过一种非技术的角度<sup>61</sup>来介绍部分运营内容。

---

<sup>52</sup> 合规有时是企业生存下去的底线，例如 P2P 清退、虚拟货币交易所的清退、金融牌照的申请更新等，目前安全负责人已经需要承担企业信息安全事故引起的法律责任。

<sup>53</sup> 不一定隶属同一个安全条线。

<sup>54</sup> 企业业务所在国家。

<sup>55</sup> 地方性合规。例如中国《数据安全法》、《个人信息保护法》(PIPL)，以及地方性的《上海市数据条例》、《深圳经济特区数据条例》、《贵州省大数据安全保障条例》等。

<sup>56</sup> 还可以划分为常规检查和临时检查。

<sup>57</sup> 安全技术承接管理团队、合规、研发等团队的需求并输出解决方案。

<sup>58</sup> 例如 DDOS 防护。

<sup>59</sup> The Resources of Vaule.

<sup>60</sup> 向自动化、精细化（场景化）、智能化的发展道阻且长。

<sup>61</sup> 这里并没有打算从生产网和办公网以及相关安全技术的角度进行介绍。

## 生命周期管理

包含对实体的申请、创建、分发、更新、轮换、吊销、删除等操作

- 证书
- 密钥
- 数据
- 漏洞
- 补丁
- 账户
- 权限
- 许可证
- 敏感信件（代表着纸媒所载的敏感信息）<sup>62</sup>

## 值守

7x24 遵循SOP的监控与响应

- 监控
- 检测
- 告警
- 应急响应

## 演练

日常熟练紧急事件下的处置流程

- 攻防对抗
- 灾难演练

## 日常<sup>63</sup>

日常工作自动化

以下按照能自动或自助与否进行划分，简单列出部分工作内容：

- 同类产品部署
- 主机/应用/网络扫描等
- 规则提取以及数据检测
- 自助服务<sup>64</sup>
  - 培训
  - 咨询<sup>65</sup>
  - 技术支持
  - 系统管理及维护
  - 编写一些制度，策略，SOP

## 管理范畴

资源需要得到合理的安排<sup>66</sup>

---

<sup>62</sup> 例如经由行政部门柜台办理所得用于下载证书及私钥的两码的纸质文件。

<sup>63</sup> daily basis.

<sup>64</sup> 一种 portal 或者本地化工具，例如证书申请、Make me admin、软件库等。

<sup>65</sup> 可以优化流程，将常见问题整理出咨询模板。

<sup>66</sup> 技术、运营、预算、人员管理等等

- 策略管理<sup>67</sup>
- 风险管理
- 生态运营<sup>68</sup>
- 项目管理
- 成本管理
- 文档管理<sup>69</sup>
- 资产管理<sup>70</sup>
- 流程管理<sup>71</sup>

## 安全框架

### SABSA<sup>72</sup>

SABSA 框架是基于风险和业务驱动，具体内容依赖标准化组织。<sup>73</sup>

Table 2: SABSA Architecture Matrix™ 2018

ASSETS (What)		MOTIVATION (Why)	PROCESS (How)	PEOPLE (Who)	LOCATION (Where)	TIME (When)
CONTEXTUAL ARCHITECTURE	Business Goals & Decisions	Business Risk	Business Meta-Processes	Business Governance	Business Geography	Business Time Dependence
	Business Value; Taxonomy of Business Assets, including Goals & Objectives, Success Factors, Targets	Opportunities & Threats Inventory	Business Value Chain; Business Capabilities	Organisational Structure & the Extended Enterprise	Inventory of Buildings, Sites, Territories, Jurisdictions, etc.	Time dependencies of Business Goals and Value Creation
CONCEPTUAL ARCHITECTURE	Business Value & Knowledge Strategy	Risk Management Strategy & Objectives	Strategies for Process Assurance	Security & Risk Governance; Trust Framework	Domain Framework	Time Management Framework
	Business Attributes Taxonomy & Profile (with integrated performance targets)	Enablement & Control Objectives; Policy Architecture; Risk Categories; Risk Management Strategies; Risk Architecture; Risk Modelling Framework; Assurance Framework	Inventory of all Operational Processes (IT, industrial, & manual); Process Mapping Framework; Architectural Strategies for IT used in process support.	Owners, Custodians and Users; Service Providers & Customers; Trust Modelling Framework	Security Domain Concepts & Framework	Through-Life Risk Management Framework; Attribute Performance Targets
LOGICAL ARCHITECTURE	Information Assets	Risk Management Policies	Process Maps & Services	Trust Relationships	Domain Maps	Calendar & Timetable
	Inventory of Information Assets; Information Model of the Business	Risk Models; Domain Policies; Assurance Criteria (populated Assurance Framework).	Information Flows; Functional Transformations; Service Oriented Architecture; Services Catalogue; Application Functionality and Services	Domain Authorities; Entity Schema; Privilege Profiles; Trust Relationship Models	Domain Definitions; Inter-domain Associations & Interactions	Start Times, Lifetimes & Deadlines
PHYSICAL ARCHITECTURE	Data Assets	Risk Management Practices	Process Mechanisms	Human Interface	Infrastructure	Processing Schedule
	Data Dictionary & Data Storage Devices Inventory	Risk Management Rules & Procedures; Risk Metadata	Working Procedures; Application Software; Middleware; Systems; Security Mechanisms; Process Control Points	User Interface to Business Systems; Identity & Access Control Systems	Workspaces; Host Platforms, Layout of Devices & Networks	Timing & Sequencing of Processes and Sessions
COMPONENT ARCHITECTURE	Component Assets	Risk Management Components & Standards	Process Components & Standards	Human Entities; Components & Standards	Locator Components & Standards	Step Timing & Sequencing Components and Standards
	Products and Tools, including Data Repositories and Processors	Risk Analysis Tools; Risk Registers; Risk Monitoring and Reporting Tools	Tools and Protocols for Process Delivery; Application Products	Identities; Job Descriptions; Roles; Functions; Actions & Access Control Lists	Nodes, Addresses and other Locators; Component Configuration	Time Schedules; Clocks, Timers & Interrupts
MANAGEMENT ARCHITECTURE	Delivery and Continuity Management	Operational Risk Management	Process Delivery Management	Governance, Relationship & Personnel Management	Environment Management	Time & Performance Management
	Assurance of Operational Excellence & Continuity	Risk Assessment; Risk Monitoring & Reporting; Risk Treatment	Management & Support of Systems, Applications & Services	Management & Support of Enterprise-wide and Extended Enterprise Relationships	Management of Buildings, Sites, Platforms & Networks	Management of Calendar and Timetable

Copyright © The SABSA Institute 1995–2018. All rights reserved.



## O-ESA

O-ESA 框架由 OpenGroup 维护，概述了企业安全体系结构的基本概念。

<sup>67</sup> 策略管理具有生命周期特性，之所以未放入运营范畴中，是因为策略更具管理性。

<sup>68</sup> 生态其实意味着背后的平台价值、平台利益。

<sup>69</sup> 包含具体的策略、制度、规范、指南、标准操作手册、事件处置流程等文档。类比 Well-Architected 应当做到 Well-Documented.

<sup>70</sup> The resources of values. 需要识别出所有具备价值的资产以及存在的潜在风险。

<sup>71</sup> 流程的制定来自不同的部门，需要遵守常规流程并设置特殊流程。例如行政流程人员离职完成之后应当立即启动 IT 流程清理遗留权限，保留关键数据等。

<sup>72</sup> <https://sabsacourses.com/wp-content/uploads/2021/02/TSI-R101-SABSA-Matrices-2018-Release-Notes.pdf>

<sup>73</sup> <https://www.secrss.com/articles/19750>

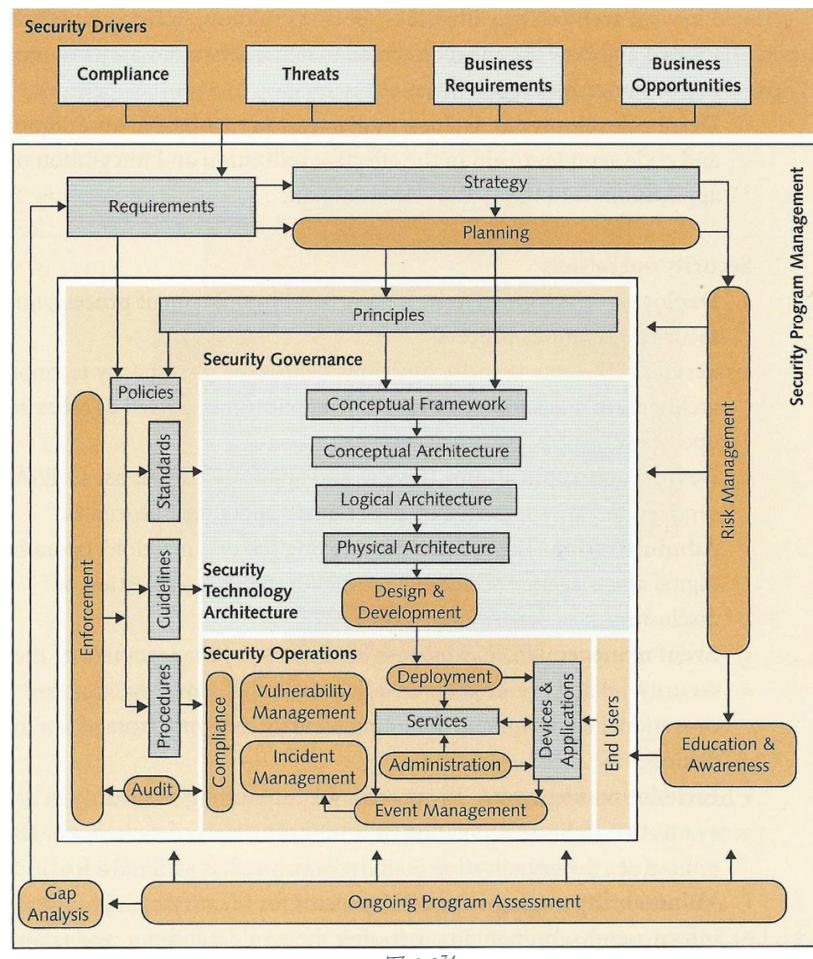


图 6-2<sup>74</sup>

## NIST CyberSecurity Framework



图 6-3

## MLPS(等级保护)<sup>75</sup>

国内用于系统防御能力级别的合规检测框架

<sup>74</sup> 图片出自 <http://www.diva-portal.org/smash/get/diva2:1031560/FULLTEXT02.pdf> 第 45 页

<sup>75</sup> [http://www.gov.cn/gzdt/2007-07/24/content\\_694380.htm](http://www.gov.cn/gzdt/2007-07/24/content_694380.htm)



图 6-4<sup>76</sup>

## 行业标准与最佳实践

行业标准提供了一种概念性的规范，企业依照自身实际情况实施形成最佳实践。设计企业安全架构时应当持续关注行业内最佳实践，一般像 AWS、Azure、Google、阿里等都会定期发布或者更新相关白皮书。例如 AWS Well-Architected Framework<sup>77</sup>：

Name	Description
<b>Operational Excellence</b>	The ability to support development and run workloads effectively, gain insight into their operations, and to continuously improve supporting processes and procedures to deliver business value.
<b>Security</b>	The security pillar describes how to take advantage of cloud technologies to protect data, systems, and assets in a way that can improve your security posture.
<b>Reliability</b>	The reliability pillar encompasses the ability of a workload to perform its intended function correctly and consistently when it's expected to. This includes the ability to operate and test the workload through its total lifecycle. This paper provides in-depth, best practice guidance for implementing reliable workloads on AWS.
<b>Performance Efficiency</b>	The ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve.
<b>Cost Optimization</b>	The ability to run systems to deliver business value at the lowest price point.
<b>Sustainability</b>	The ability to continually improve sustainability impacts by reducing energy consumption and increasing efficiency across all components of a workload by maximizing the benefits from the provisioned resources and minimizing the total resources required.

图 6-5

以及 Microsoft Cybersecurity Reference Architectures<sup>78</sup>：

<sup>76</sup> [https://pdf.dfcfw.com/pdf/H3\\_AP202107301506980769\\_1.pdf?1627652267000.pdf](https://pdf.dfcfw.com/pdf/H3_AP202107301506980769_1.pdf?1627652267000.pdf)

<sup>77</sup> 关注行业最佳实践时不应仅止步于安全范围

<sup>78</sup> <https://docs.microsoft.com/en-us/security/cybersecurity-reference-architecture/mcra>

## Microsoft Cybersecurity Reference Architectures (MCRA)

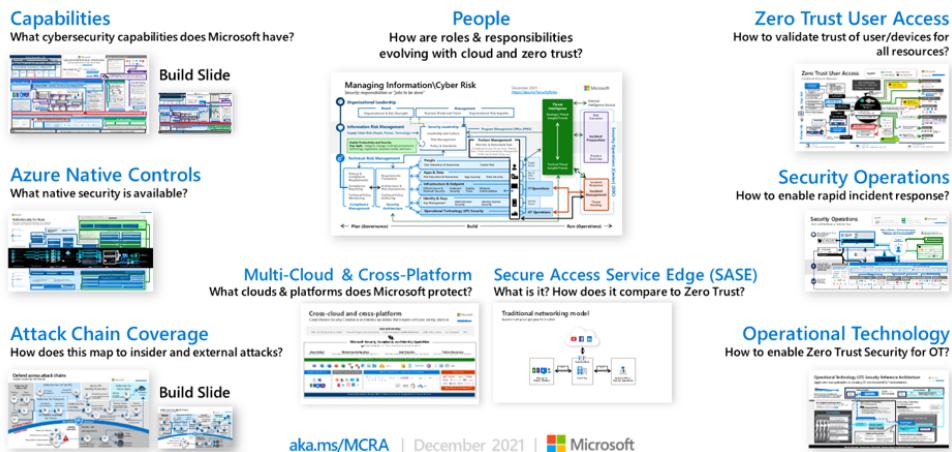


图 6-6

## 上云与云上

### 讨论云

上云<sup>79</sup>必然是趋势，对于一些行业却因为合规原因无法使用。CSP<sup>80</sup>固然能够提供基础设施层面的默认安全，但对于应用、服务、权限以及数据层面的威胁仍需要时刻关注平台安全。期待密码学的发展能够为租户带来更高的数据安全体验。

### 讨论云原生<sup>81</sup>

云上将快速敏捷的持续交付作为云原生的目标，其中所涉及的流程及基础设施整体如果同安全能力<sup>82</sup>不相匹配，只会导致暴露更多更被动的攻击面。

<sup>79</sup> 意味着采用云计算技术对底层资源实现弹性调度，但并不意味着使用 CSP 的服务，也可以是自建。

<sup>80</sup> Cloud Service Provider

<sup>81</sup> <https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/definition>

<sup>82</sup> 目前有针对云原生提出安全切面的概念

# 第七章：安全架构基础

本章通过介绍安全架构以便达到能够对企业架构目标进行拆分或组合，并使其具备安全特性的能力。

工欲善其事，必先利其器<sup>83</sup>。

——《论语》

## 定义

提供一种供企业实现架构安全特性的一系列原则、流程等组合方式。

## 目标

### 关注价值与威胁<sup>84</sup>

包含价值所在以及来自内外部的威胁<sup>85</sup>

- 确定当前价值<sup>86</sup>
- 确定风险与差距<sup>87</sup>
- 确定优先级<sup>88</sup>

### 关注基础能力建设

包含技术、运营、管理三个基本面的能力建设<sup>89</sup>

- 用于技术支撑的安全基础架构<sup>90</sup>
- 梳理资产及资产属性<sup>91</sup>
- 基本策略和过程控制<sup>92</sup>

### 关注生命周期治理

包含安全治理中各方面生命周期的运营管理<sup>93</sup>

---

<sup>83</sup> 凡有形有象者即器也，所以为器者所以为器之理者即道也——朱熹。

<sup>84</sup> 持续关注，持续优化。

<sup>85</sup> 思考已知的威胁还是未知的威胁？

<sup>86</sup> 现在业务要做什么？安全要做什么？业务能带来什么价值？安全能为业务带来什么什么样的价值？

<sup>87</sup> 面临的风险和现实的差距

<sup>88</sup> 思考是应该先处理重要且紧急的呢还是重要不紧急的呢？

<sup>89</sup> 组织架构应当能够支撑基础能力建设。

<sup>90</sup> 各种工具、系统、软硬件等，用于帮助其他系统减缓入侵。

<sup>91</sup> 硬件资产、应用资产、数据类别、重要程度、用途、所属等。

<sup>92</sup> 各个领域（对用户、应用、权限、数据等）的 security control 以及对应的标准操作流程等。

<sup>93</sup> 可与第六章中运营部分生命周期管理部分互相参考

- 引用业界参考模型<sup>94</sup>以及确定适用的管理阶段<sup>95</sup>
- 追踪不同阶段的治理效果<sup>96</sup>
- 采用平台或者自动化工具<sup>97</sup>

## 安全模式<sup>98</sup>

为系统<sup>99</sup>架构附加安全特性设计模式

- 代理模式：Inbound/Outbound Proxy
- 分层模式：Data Access Layer, Traffic Access Layer, ORM.
- 聚合模式：API Gateway
- 分离模式：Sanbox

## 安全服务<sup>100</sup>

引入并对外提供安全服务

可以是提供产品<sup>101</sup>出去，亦或是产品提供的服务，亦或人工服务，例如技术支持，客户咨询等。针对安全服务需要另从雇佣关系上需要注意外包人员的管理，尤其是“外包的外包”，例如 x 公司作为许多企业的外包服务提供者，其自身还会雇佣外包为雇主企业提供服务。

- 基础运维<sup>102</sup>：Logging & Monitoring, Auditing, Administration, Alerting, Telemetry, NTP, Password Management, User Role, Port(eg. baseline 443, 2222) ACL etc;
- 技术支持：Application Support, IT Support, Technical Support, Architecture Review, Code Reveiew, Penetration testing etc;
- 意识培训：Awareness Training, Anti-Corruption Training, Security Development etc;
- 定制开发：Service Integration, Self Service<sup>103</sup>, Security SDK etc;
- 客户咨询：Customer Service, Security Solution;

针对提供的服务应通过确定衡量指标<sup>104</sup>、收集服务反馈以达到对服务质量<sup>105</sup>管理以及优化用户体验。

<sup>94</sup> 引用参考框架或模型帮助识别资产种类以及管理周期。例如 IPDRR（防御流程），DSMM（数据管理），CVSS（漏洞识别），ATT&CK（攻击识别），STRIDE（威胁识别）等。

<sup>95</sup> 假设某些存证要求 5-10 年保留的话，短期内是不需要投入到数据销毁上的。不同实体对应的生命周期也不相同，并非所有管理阶段都需要覆盖。例如仅需要从收集、传输、存储、处理、共享、销毁当中根据实际情况选择固定的阶段进行处置即可。

<sup>96</sup> 确保落地效果，以及互相监督。例如密钥的流转及销毁。

<sup>97</sup> 用技术去实现过程控制和流程管理，可以减少人工失误并且易于审计。

<sup>98</sup> 原英文为 Security Patterns. 这里仅仅列举几条。有的工具可以支持多种模式。详细可参考 Security Patterns In Practice - Designing Secure Architectures Using Software Patterns 以及 OSA 文档库。

<sup>99</sup> 亦有为安全开发提供的安全模式。

<sup>100</sup> 包含工具，服务，产品等。可以通过采购、自研、开源社区等引入并以手动或自动的形式对内外部提供。具体类别可参考第六章中的技术范畴、运营范畴、管理范畴三部分。

<sup>101</sup> 借助 vendor 技术优势往往存在新的问题。例如乙方的自我假设，以及特定的场景问题，通常还需要访问外网进行数据同步（系统升级，特征库更新等）。

<sup>102</sup> 安全工具的管理平台一般是 B/S 架构，端上组件是 C/S 结构。基础运维需要对相关系统的日志、审计，访问控制、权限控制、监控、告警等一系列选项进行配置维护。另注意在此过程中引入了信任的第三方。例：引入 snmp 的话，意味着对 snmp server 的信任。Hids Agent 对 Server 亦类似。

<sup>103</sup> 自一般情况下，自助服务属于 L0 Support. 客户咨询属于 L1 Support.

<sup>104</sup> 确定采用哪些 metric 衡量服务质量。例如 SLA 衡量服务可用性；

<sup>105</sup> 提高服务质量可参考 PDCA 循环。

# 第八章：解决方案<sup>106</sup>

本章介绍如何去寻找一个合适的度<sup>107</sup>，去设计简单的解决方案<sup>108</sup>以及相关内容模块。

如切如磋，如琢如磨<sup>109</sup>。

——《诗经》

## 明确需求<sup>110</sup>

期望的范围（原始需求的目标期望）

详细描述方案的原始需求，并给出目标期望。衡量自己的资源，是否要削减非必要需求或者延缓低优先级需求，确保能够在当前的资源条件下进行实施。例如可以将需求分阶段实施，一期去实行紧急且重要的需求。

## 适用场景<sup>111</sup>

需求的范围（基础设施、应用、数据等）

原始的需求结合当前的实际情况确定出有哪些适用的场景<sup>112</sup>。例如进行商密改造（负载均衡、API 网关、Web 服务器、应用服务、密码学库等）；Ipv6 改造（线路、路由、安全防护、DNS 服务器、应用服务等）；代码扫描（发布系统、源代码控制系统等）；流量监控（多端、进出口流量等）；

## 需求依赖<sup>113</sup>

风险与资源的范围（预算、人力、技术、项目管理等）

需要什么资源（预算、采购、周期、关键人员、关键技术等），有哪些已知的风险（资源不足、遗留系统、整改有期限等）。衡量已知的风险和实现需求目标的资源，判断是否能够解决，或者是需要寻求哪些进一步的支持。

## 最佳实践<sup>114</sup>

参考答案的集合（需求、场景、资源、风险等）

---

<sup>106</sup> 不一定是安全问题的解决方案架构师，安全架构师也需要能够输出解决方案。

<sup>107</sup> 权衡，适度。

<sup>108</sup> 解决方案内容需要不同视角支撑、包含业务、技术、功能、实施等。

<sup>109</sup> 诗经中本用于形容君子的自我修养，解决方案也应当根据需求精细打磨。没有充足的理由下当拒绝去使用临时方案。

<sup>110</sup> 安全的需求可能来自法务（合规）、产研（技术）、管理等。明确需求主要是用于确定目标的期望。

<sup>111</sup> 适用场景其实是实际情况中的具体需求。

<sup>112</sup> 例如下文提到的商密和 Ipv6 改造一期可能是仅支持网络层入口流量。

<sup>113</sup> 需求资源支持，明确职责范围。

<sup>114</sup> 通过组合工具、服务及相关资源去解决特定的需求问题。理论上需要设计 2-3 种具有明显差异化的架构方案进行对比较为合适，以此选择出最佳实践。

## 方案设计

包含<sup>115</sup>技术架构、流程、运营等，是逻辑架构到上下文架构的具体体现<sup>116</sup>。通过基础设施的组合实现技术方案<sup>117</sup>，同时提供日常操作、维护、特例等处置的参考案例。

## 验收规范

通过一定的测试用例来判断是否符合目标期望<sup>118</sup>，以及能否维持 SLA，具备 SOP<sup>119</sup>，符合验收规范后可以进行 Sign Off。

## 参考架构

提供行业标准、行业最佳实践的相应说明以作参考。

## 项目管理

明确范围，减少待定的过程

从明确需求到具体的适用场景，以及梳理依赖到输出最佳实践的过程都离不开项目管理。上至寻求支持，下至任务分配。即便在整个过程中存在专业的项目经理时，作为安全架构师还是应该能够完成跨团队沟通，通过会议、邮件、IM 等形式确定好关键利益人(Stakeholder)、接口人等。简单的来说，务必要求对每一个输入<sup>120</sup>都有一个输出以及减少输出中的待定项。

---

<sup>115</sup> 建议包含，不意味着一定全部具备。根据实际场景而言，大多只包含了部分。

<sup>116</sup> 从 Conceptual Architecture 到 Contextual Architecture，从 High Level Design 到 Low Level Design。

<sup>117</sup> 是否引入了新的模块，新的模块是否需要定制化亦或完全自研，是否符合前文中的架构基础。例如具备监控告警，符合 HA 之类的特性。

<sup>118</sup> 适用场景的具体需求有没有得到解决，是否符合预期。

<sup>119</sup> 以及其他过程中的相关文档（Policy、Guideline 等）。文档化是架构设计过程中的一个主要输出。

<sup>120</sup> 多为项目级别的输入，这些输入进行评估，判断得到一个结论作为输出，有时候可能没有得到结果是待定（TBD），但应该逐步去减少结论中的待定项。例如需要多久进行定期的 review，找到谁能够给到支持，是否必要采购什么产品等。

# 第九章：持续服务<sup>121</sup>

方案<sup>122</sup>实施包含了部署，技术支持，客户咨询等方面。这意味着在方案落地之后项目只是达到了阶段性交付，并未结束，依旧需要持续运营。本章通过介绍实施过程中的相关问题以及持续运营来认识持续交付。

凡事豫则立，不豫则废。

——《礼记·中庸》

## 以客户为中心

理解每个客户<sup>123</sup>的差异性，统一基础上提供定制化服务，并通过 PDCA 持续改进。

## 部署

结合基础设施完成解决方案的部署并满足架构质量<sup>124</sup>。例如采用负载均衡、完成域名申请并增加相应 DNS 解析、开通防火墙规则、接入日志平台、增加监控告警、设置权限分组等。整体大致可分为初始化配置<sup>125</sup>、上线测试、对外服务、系统维护等。

## 运营

持续服务的主要体现就是运营，主要包含了日常操作<sup>126</sup>、客户咨询、技术支持、专项治理等；

## 交付<sup>127</sup>

基于已有方案，根据不同的需求交付新的功能。

## 系统集成

集成其他服务，以及被其他服务集成。例如：Vault 和 Nessus、Aqua 集成；HSM 和 KMS、AD、PKI 集成；NTP、Zabbix 同许多基础设施集成等；

## 定制开发

集成、部署、运营的过程均可将日常工作自动化。例如自助服务（代码扫描、证书申请等），定制服务（扩展功能、支持新的平台等）；

---

<sup>121</sup> 借用敏捷思想进行持续的交付、集成、部署等。安全也要提供持续服务。

<sup>122</sup> 交付的方案只是当前（中期）最适合的，并不一定是最优的（受限于资源），仍需持探索更新。

<sup>123</sup> 首先要明确所在安全部门的客户有哪些？产研、兄弟团队、法务等。

<sup>124</sup> 参考第五章中架构质量相关内容。

<sup>125</sup> 需符合基础设施基线原则，即结合基础设施，可以超过基础设施安全标准，但不可低于。

<sup>126</sup> 应当尽量做到自动化，或者采用工具辅助重复性的运营工作。

<sup>127</sup> 在合理情况（同类产品同类功能）下扩大支持的功能范围。

# 第十章：成为安全架构师

总有猎头问候选人：“你带过团队吗？”，以此判断对方是否具备管理经验。但实际上并不是去打 KPI 才会管理<sup>128</sup>，同样的也并不是只有架构师的角色才会承担安全架构的设计。本章简单介绍一些心得，重在持之以恒。

骐骥一跃，不能十步；驽马十驾，功在不舍。<sup>129</sup>

——《荀子·劝学》

## 品格

拒绝贪腐<sup>130</sup>，严守底线；

## 心态

戒急<sup>131</sup>、戒忿<sup>132</sup>、戒贪<sup>133</sup>、持续学习、虚怀若谷；

## 健康

身体健康最重要，照顾好自己、家庭，保持健康的身体有利于思维创新；

## 技能<sup>134</sup>

主要包括对技术，运营，管理等各方面的宏观认识<sup>135</sup>，以及对具体方案的细节把握。持续研究行业最佳实践，寻找机会落地<sup>136</sup>；

## 知行<sup>137</sup>

有知有行，知行合一；

---

<sup>128</sup> 国企内部一般以虚线带人开始培养其向管理方向发展。

<sup>129</sup> 蝇无爪牙之利，筋骨之强，上食埃土，下饮黄泉，用心一也。蟹六跪而二螯，非蛇鳝之穴无可寄托者，用心躁也。

<sup>130</sup> 不要说什么面对诱惑经受过诱惑才是真的做到了。当面对诱惑却并不知道是否抵住诱惑时，那就应该坚决的拒绝这种诱惑。

<sup>131</sup> 遇事着急容易坏事，三思后行。急躁也容易导致身体不健康。

<sup>132</sup> 责人之心责己，恕己之心恕人。知易行难，笔者时时对一些事情也无法容忍。

<sup>133</sup> 贪多嚼不烂。

<sup>134</sup> Business is Business.

<sup>135</sup> 此时不求处处精通，但必须知道哪个地方有个“开关”，按下去有什么效果。

<sup>136</sup> 纸上得来终觉浅，绝知此事要躬行。

<sup>137</sup> 知道自己要做的事情，也知道自己的屁股在哪里。

# 附录. 推荐阅读<sup>138</sup>

1. AWS 安全最佳实践: <https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/wellarchitected-security-pillar.pdf>
2. AWS Well-Architected Framework: <https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome.html>
3. Google Infrastructure Security Design: <https://cloud.google.com/security/infrastructure>
4. Microsoft Cybersecurity Reference Architecture: <https://docs.microsoft.com/en-us/security/cybersecurity-reference-architecture/mcra>
5. Microsoft SDL: <https://www.microsoft.com/en-us/securityengineering/sdl>
6. OpenSecurityArchitecture Pattern Library: <https://www.opensecurityarchitecture.org/cms/library/patternlandscape>
7. A Detailed guide on building your own pki: <https://www.encryptionconsulting.com/a-detailed-guide-on-building-your-own-pki/>
8. Secret Management Best Practice: <https://zhaokunpeng.medium.com/hashicorp-vault-advanced-tutorial-for-enterprise-56223aae39bb>
9. Cloud Security and DevSecOps Best Practices by SANS: <https://sansorg.egnyte.com/dl/OTrn2Tyq3n>
10. 零信任架构综述: [https://mp.weixin.qq.com/s/yQIZKFBQuNuTzvy-e1ZI\\_g](https://mp.weixin.qq.com/s/yQIZKFBQuNuTzvy-e1ZI_g)
11. 全面解决零信任安全架构 by 奇安信: <https://www.gartner.com/teamsiteanalytics/servePDF?g=/imagesrv/media-products/pdf/Qi-An-Xin/Qi-An-Xin-1-1OKONUN2.pdf>
12. 数据安全技术与产业发展研究报告 2021: [http://www.caict.ac.cn/kxyj/qwfb/ztbz/202112/t20211221\\_394364.htm](http://www.caict.ac.cn/kxyj/qwfb/ztbz/202112/t20211221_394364.htm)
13. 隐私计算白皮书: <https://gw.alipayobjects.com/os/bmw-prod/73c5f163-d091-487a-bf5c-41841f546bc0.pdf>
14. 解构安全运行能力体系建设: <https://www.qianxin.com/topic/rsac/newsDetails?id=40>
15. 《Enterprise Security Architecture》 - Nicholas Sherwood
16. 《Designing Security Architecture Solutions》 - Jay Ramachandran
17. 《Security Patterns In Practice - Designing Secure Architectures Using Software Patterns》 - Eduardo Fernandez
18. 《Practical Cybersecurity Architecture》 - Ed Moyle, Diana Kelley
19. 《Hands-on cybersecurity for architects plan and design robust security architectures》 - Aslaner, MiladRerup, Neil
20. 《互联网企业安全高级指南》
21. 《大型互联网企业安全架构》
22. 《Google 系统架构解密》
23. 《架构整洁之道》
24. 《代码整洁之道》
25. 《Getting Real》
26. 《Web 信息架构》
27. 《企业应用架构模式》
28. 《网络安全法——网络安全等级保护 2.0》
29. 《微服务设计》
30. 《高可用架构》

---

<sup>138</sup> 顺序无先后，应读当尽读。