塔防在私有云安全中的实践

黄晟

提纲

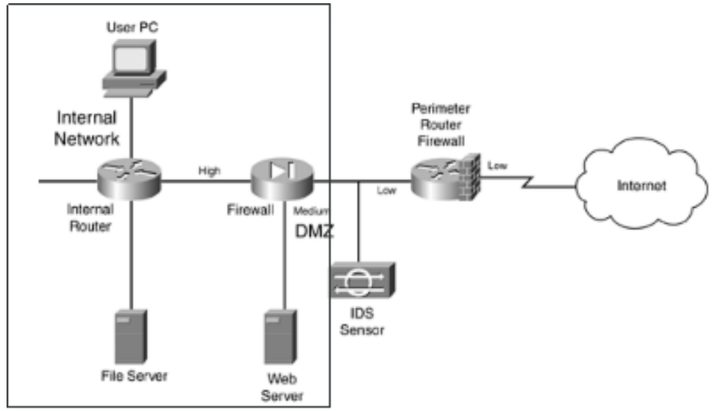
- “塔防”思路的回顾
- 面向企业私有云安全的纵深防御实践
- “塔防”纵深防御实践的展望

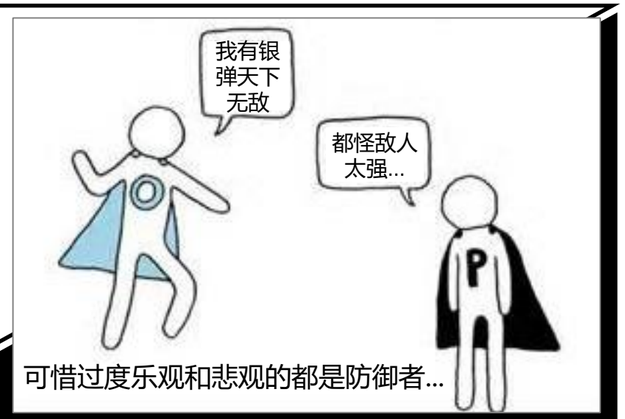
Part 01

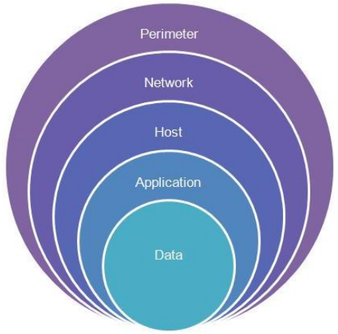
“塔防”思路回顾

“塔防”游戏对网络安全纵深防御的启迪

为啥会“瞎琢磨”纵深防御?


User PC
Internal Network
Internal Router
High
Firewall
Medium
DMZ
Low
Perimeter Router Firewall
Low
Internet
IDS Sensor
File Server
Web Server

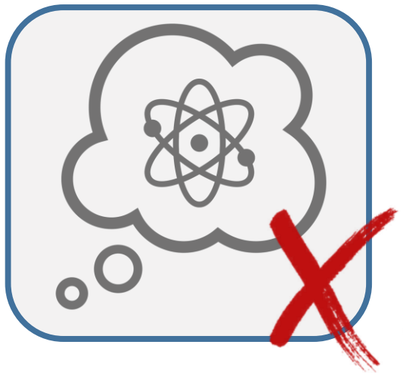

我有银弹天下无敌
都怪敌人太强...
可惜过度乐观和悲观的都是防御者...

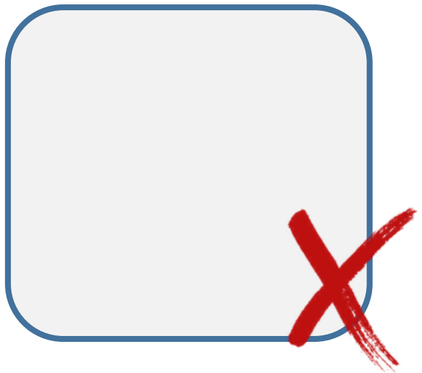

Perimeter
Network
Host
Application
Data

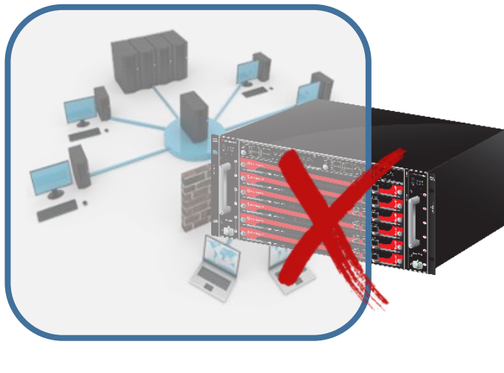
KNOWN VULNERABILITIES
UNKNOWN VULNERABILITIES

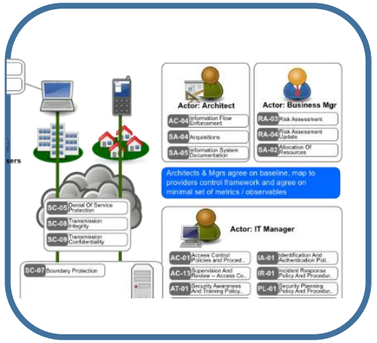
“塔防”是什么……

- “塔防”不是一种理论；
- “塔防”不是一个产品；
- “塔防”不是一个不变的解决方案；
- “塔防”一开始是一个比喻；
- “塔防”可能算一种思路；
- “塔防”可能会形成一系列模式；
- 构建有效的纵深防御体系才是目标！









复杂攻击与精妙利用层出不穷—绝望的防守者?

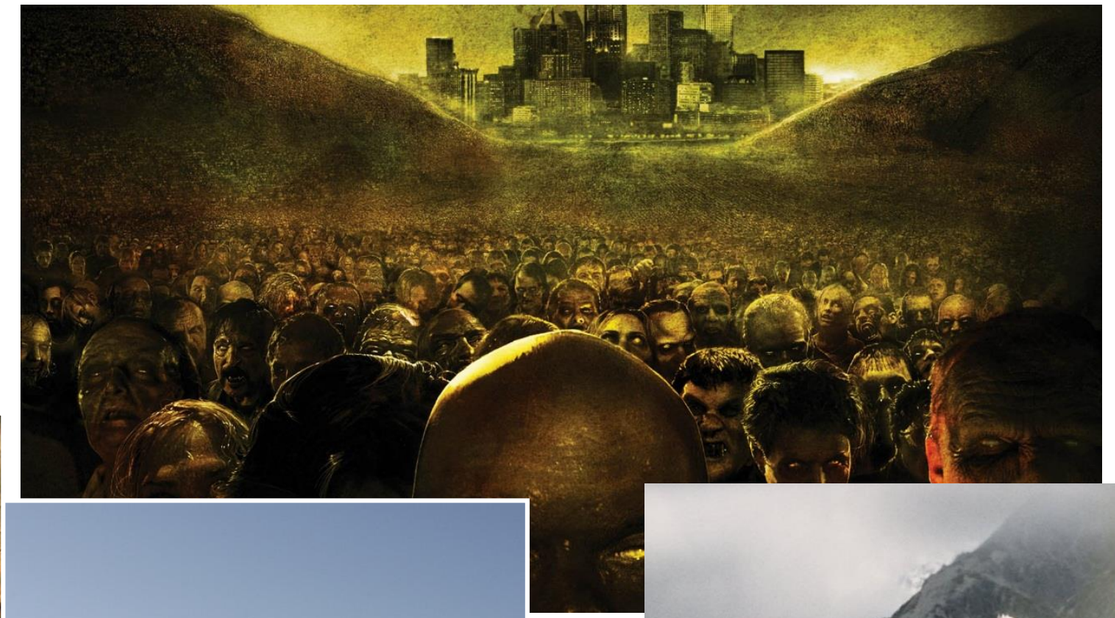
信息安全防御变得越来越困难，感觉不管采用怎么强悍的防御手段，都很快会被一些“精妙”的攻击所击破：

- APT攻击让传统防御手段变得形同虚设；
- 信息交互的刚需使网络隔离难以奏效；
- 各种宣称“解决一起安全问题”的防御技术很快被绕过...

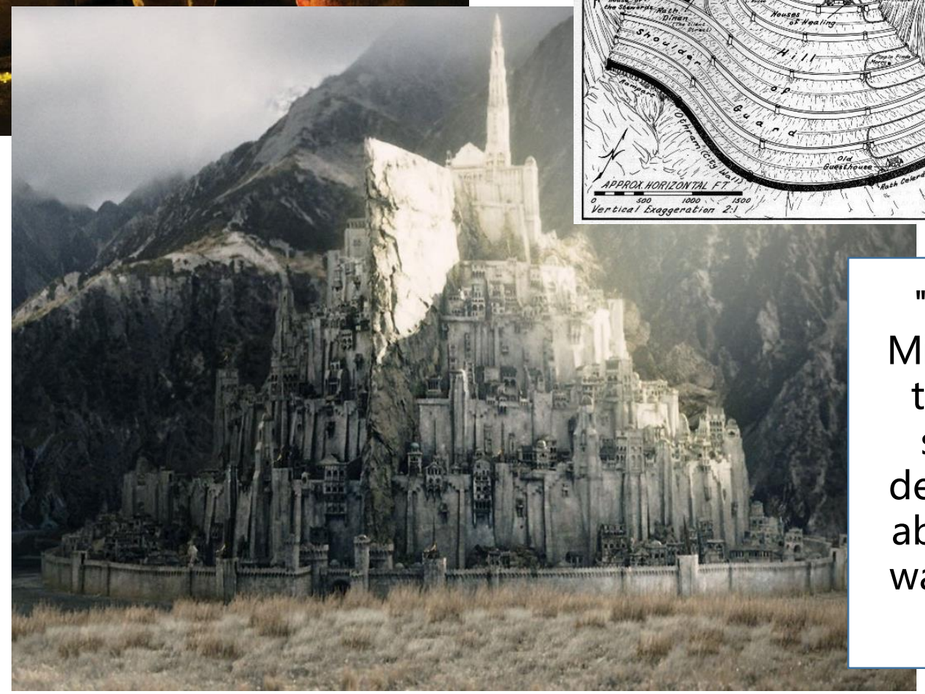
信息安全防御看起来那么让人绝望，而防御者看起来是那么地...

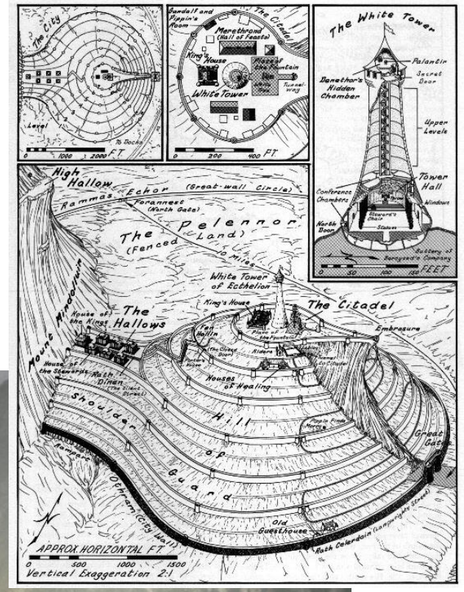
更糟糕的是，防御者需要保护的信息系统就像游戏中的BOSS那样，最终逃不过被打垮的命运...

如何面对“丧尸式”的进攻...



V
S





"For the fashion of Minas Tirith was such that it was built on seven levels, each delved into a hill, and about each was set a wall, and in each wall was a gate."

对攻与防的思考 — 是不是“玩错游戏”了?

如果按照玩“英雄冒险”游戏的方式来考虑信息安全攻防对抗，可能信息安全防守真是令人绝望的... 防守者不可能抛弃待保护的资产而全面出击！在此模式下，防守方者面对专注进攻的对手缺少优势!

但是，假如防守者抛弃“英雄对决”的幻想，利用防守者的先发优势，转而选择“塔防”的“游戏模式”，依托纵深防御对抗攻击者，情况会改变吗?

对防御者而言，他会放弃在领土上相对微弱的抵抗，而采取全力压迫攻击方之后勤补给，或是切割敌方在数量上的优势的兵力。一但攻击方失去其动能，或是其在大部份地区被切割后的兵力数量优势不再，防御反攻将在敌人虚弱地带发动，其主旨在于促使敌方资源之消耗，进而带动消耗战，或是迫使攻击方退回原本攻击起始点。

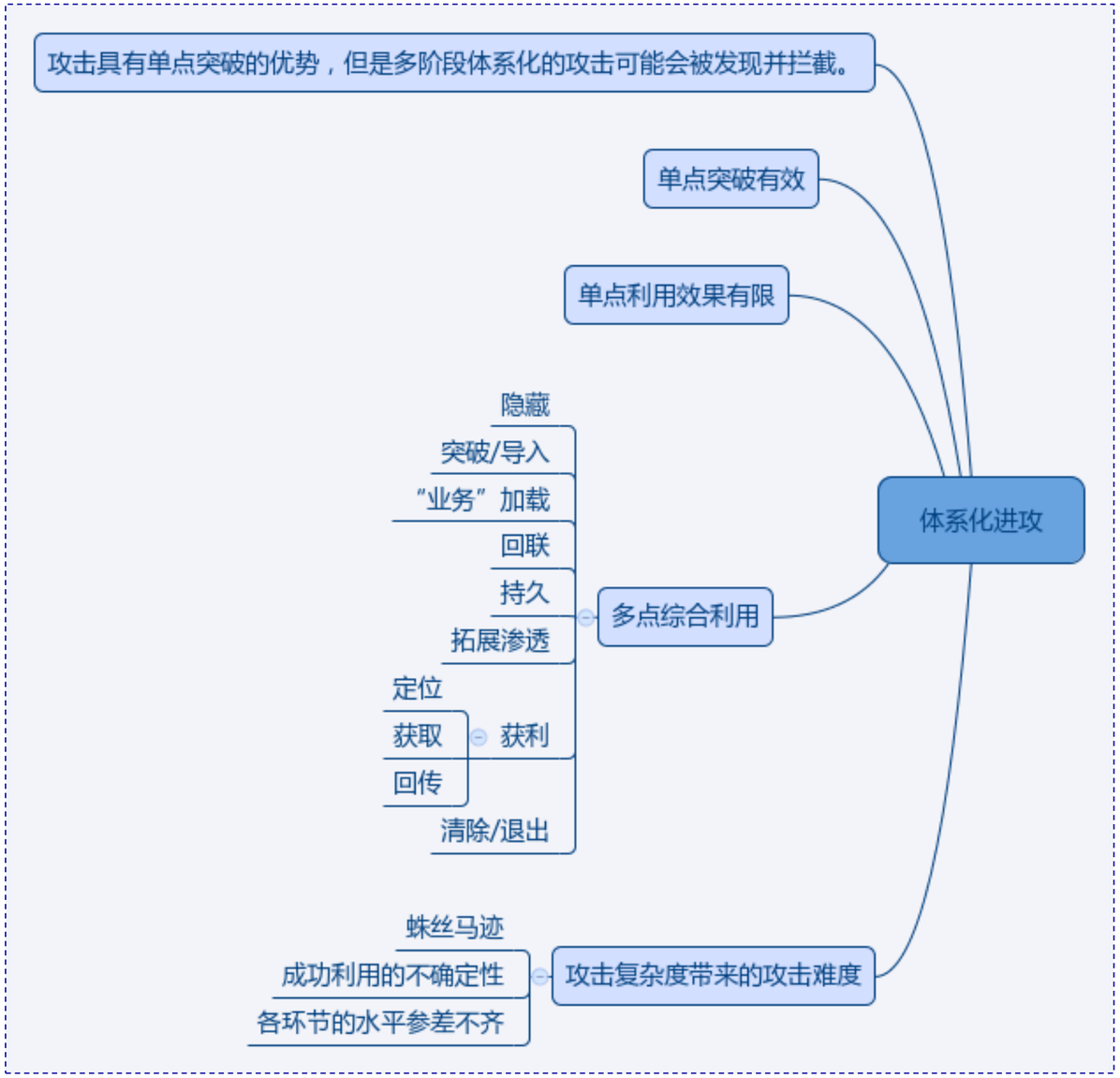
—— 摘自Wikipedia的“纵深防御”词条

塔防游戏里的先发优势

正视严峻挑战 — 体系化攻击带来的严重威胁

信息安全漏洞层出不穷，而且利用手法也日益精妙，但是这些突破大多都是“单点突破”，而绝大多数一个单点突破自身都难以达到攻击目的。

因此目前可以观测到的趋势正是攻击行为的“体系化”，攻击阶段越来越多，甚至攻击任务都出现了“专业外包”的情况。面对这样复杂的进攻，传统的安全边界或网络隔离策略难以奏效...

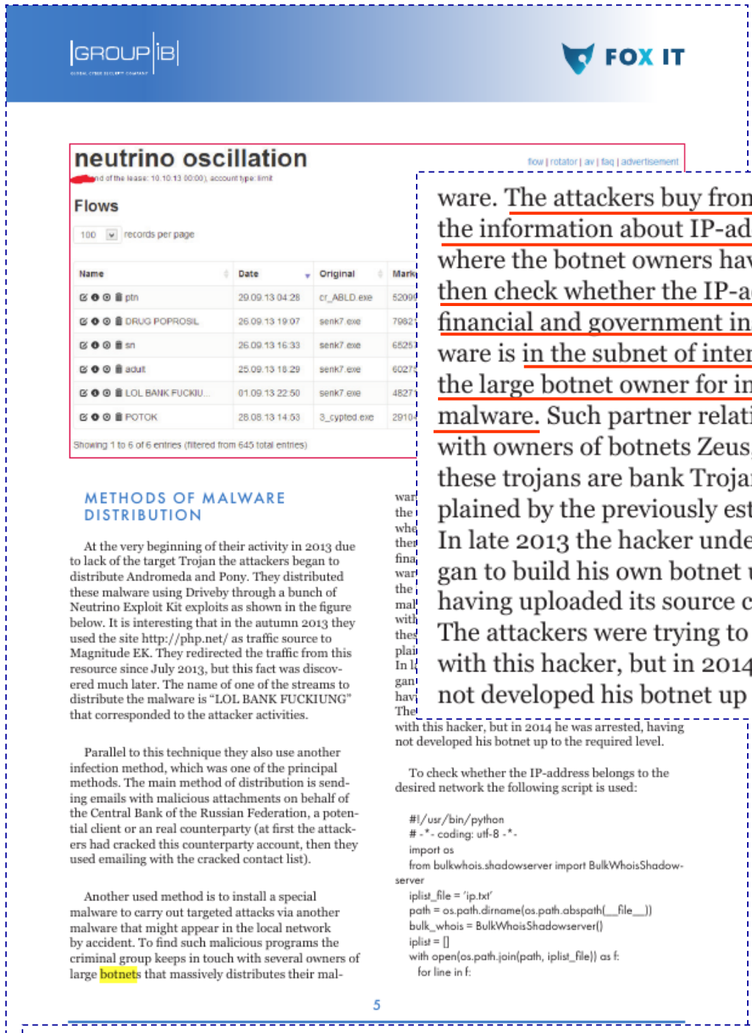


体系化的攻击——“业务化”的“攻击供应链”

俄罗斯犯罪团伙恶意代码Anunak开展对金融机构开展APT攻击，在Group-IB与Fox-IT 的联合分析报告中，可以发现一个很有意思的细节，那就是对“攻击供应链”的“业务化”利用。

攻击者与大型僵尸网络“运营商”形成了紧密的合作关系，攻击者不是简单地将恶意代码交给“运营商”大规模散布，而是在“运营商”提供的肉鸡名单中进行IP地址比对挖掘，找出那些可能属于金融机构或政府部门的肉鸡，并指示“运营商”向这些特定目标投放恶意代码。

看看咱们内网的僵尸网络肆虐情况，“细思极控”啊！


GROUP-IB

FOX IT

neutrino oscillation

Flows

Name	Date	Original
ptn	29.09.13 04:28	cr_ABLD.exe
DRUG POPROSIL	26.09.13 19:07	senk7.exe
sn	26.09.13 16:33	senk7.exe
aout	25.09.13 18:29	senk7.exe
LOL BANK FUCKIU...	01.09.13 22:50	senk7.exe
POTOK	28.08.13 14:53	3_cypted.exe

Showing 1 to 6 of 6 entries (filtered from 645 total entries)

METHODS OF MALWARE DISTRIBUTION

At the very beginning of their activity in 2013 due to lack of the target Trojan the attackers began to distribute Andromeda and Pony. They distributed these malware using Driveby through a bunch of Neutrino Exploit Kit exploits as shown in the figure below. It is interesting that in the autumn 2013 they used the site http://php.net/ as traffic source to Magnitude EK. They redirected the traffic from this resource since July 2013, but this fact was discovered much later. The name of one of the streams to distribute the malware is “LOL BANK FUCKIUNG” that corresponded to the attacker activities.

Parallel to this technique they also use another infection method, which was one of the principal methods. The main method of distribution is sending emails with malicious attachments on behalf of the Central Bank of the Russian Federation, a potential client or an real counterparty (at first the attackers had cracked this counterparty account, then they used emailing with the cracked contact list).

Another used method is to install a special malware to carry out targeted attacks via another malware that might appear in the local network by accident. To find such malicious programs the criminal group keeps in touch with several owners of large botnets that massively distributes their malware. The attackers buy from these botnet owners the information about IP-addresses of computers where the botnet owners have installed malware and then check whether the IP-address belongs to the financial and government institutions. If the malware is in the subnet of interest, the attackers pay the large botnet owner for installation of their target malware. Such partner relations were established with owners of botnets Zeus, Shiz Ranbyus. All of these trojans are bank Trojans, their usage is explained by the previously established relationships. In late 2013 the hacker under the alias Dinhold began to build his own botnet using modified Carberp, having uploaded its source code for public access. The attackers were trying to create similar relations with this hacker, but in 2014 he was arrested, having not developed his botnet up to the required level.

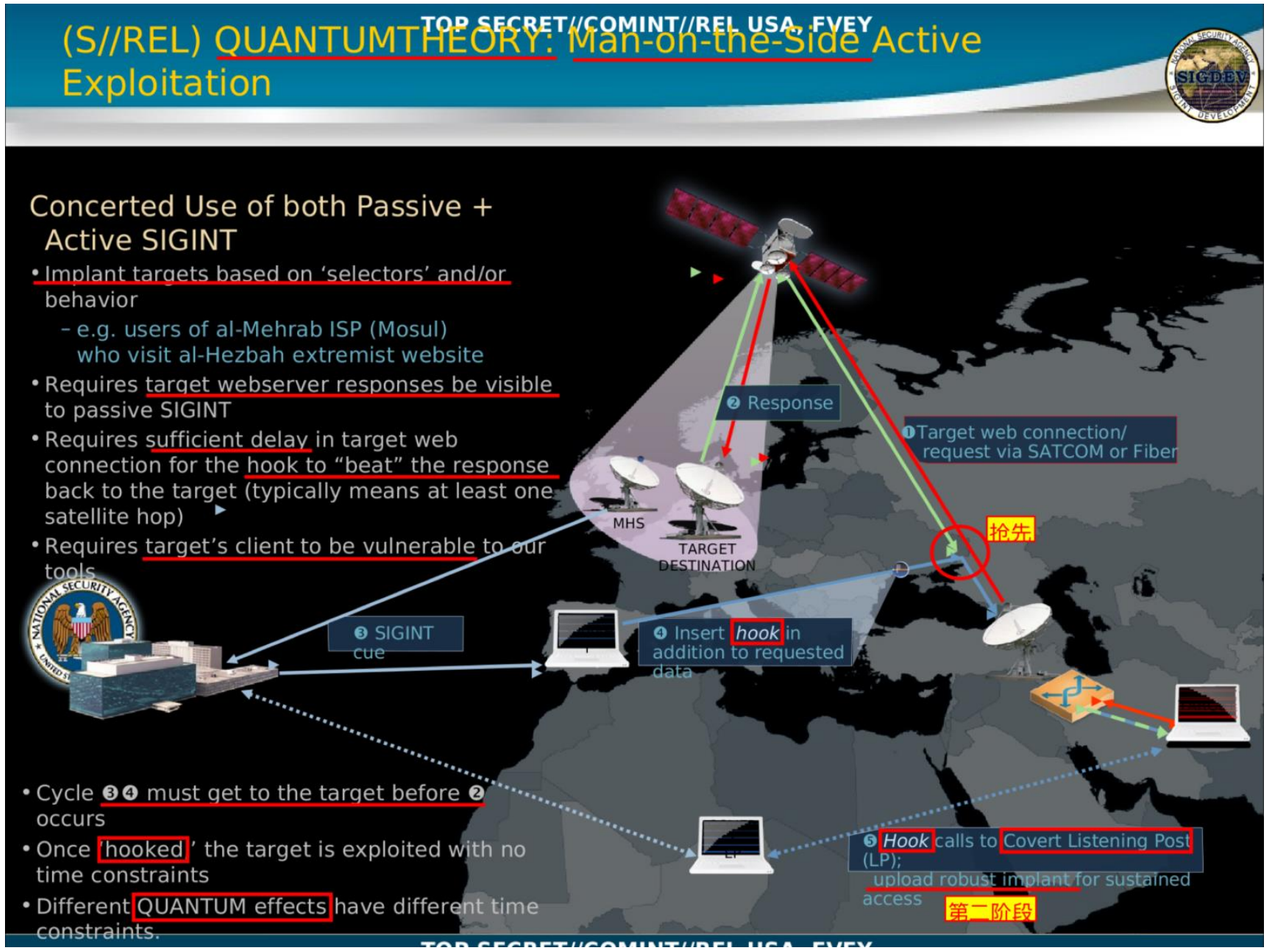
To check whether the IP-address belongs to the desired network the following script is used:

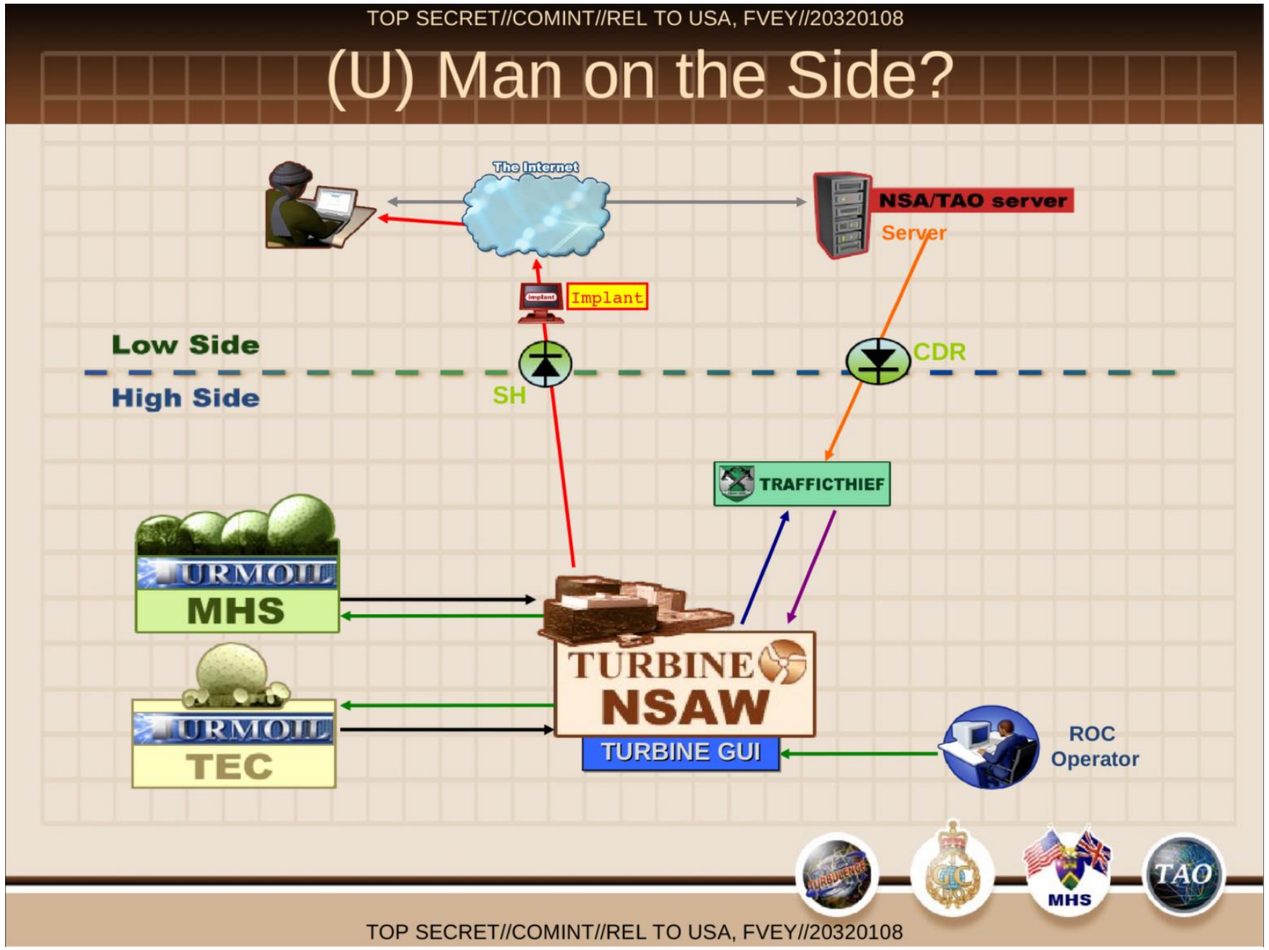
```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import os
from bulkwhois.shadowserver import BulkWhoisShadowserver
iplist_file = 'ip.txt'
path = os.path.dirname(os.path.abspath(__file__))
bulk_whois = BulkWhoisShadowserver()
iplist = []
with open(os.path.join(path, iplist_file)) as f:
    for line in f:
```

5

https://www.fox-it.com/en/files/2014/12/Anunak_APT-against-financial-institutions2.pdf

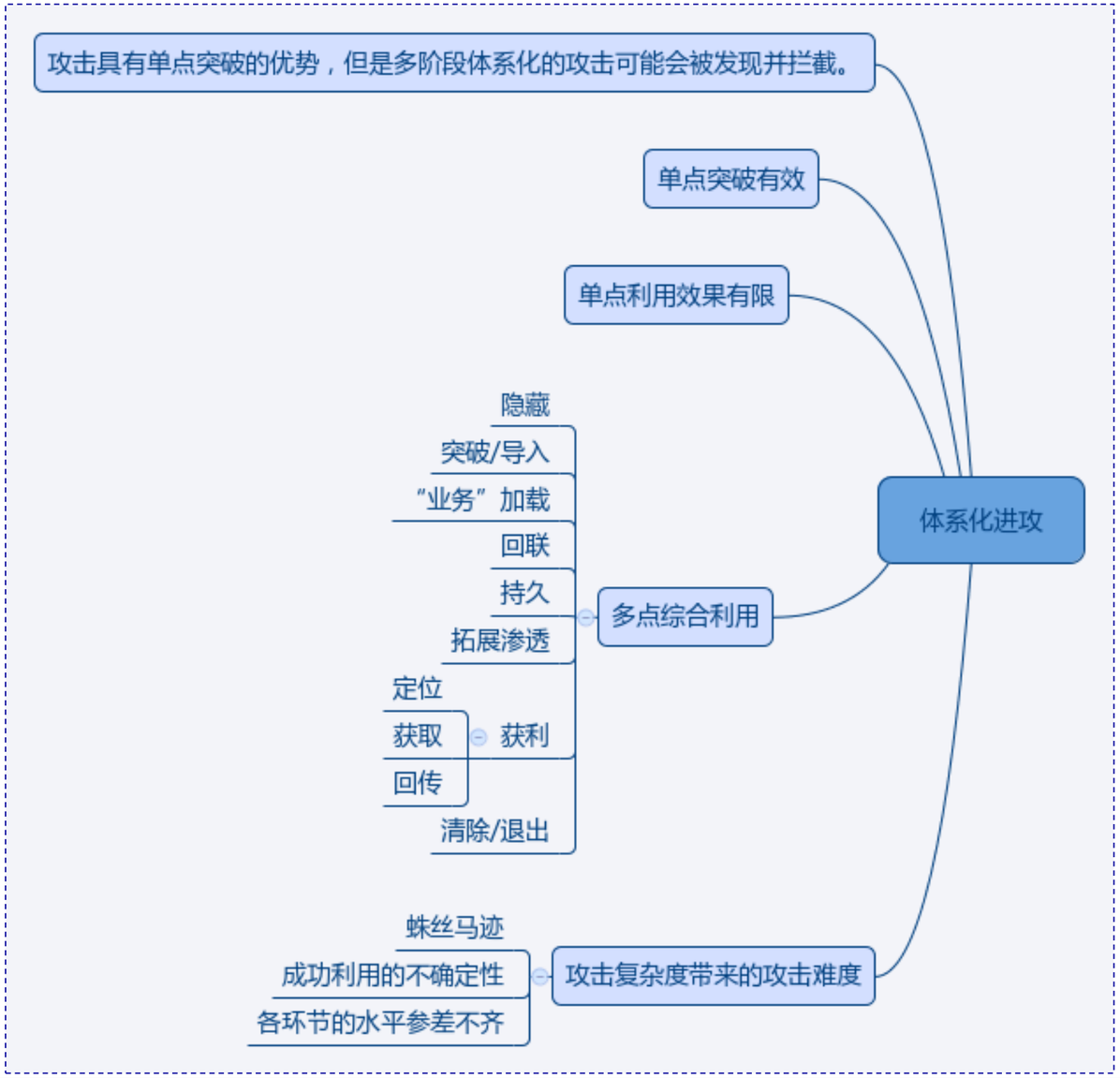
体系化的进攻 —— NSA的“工程奇迹”



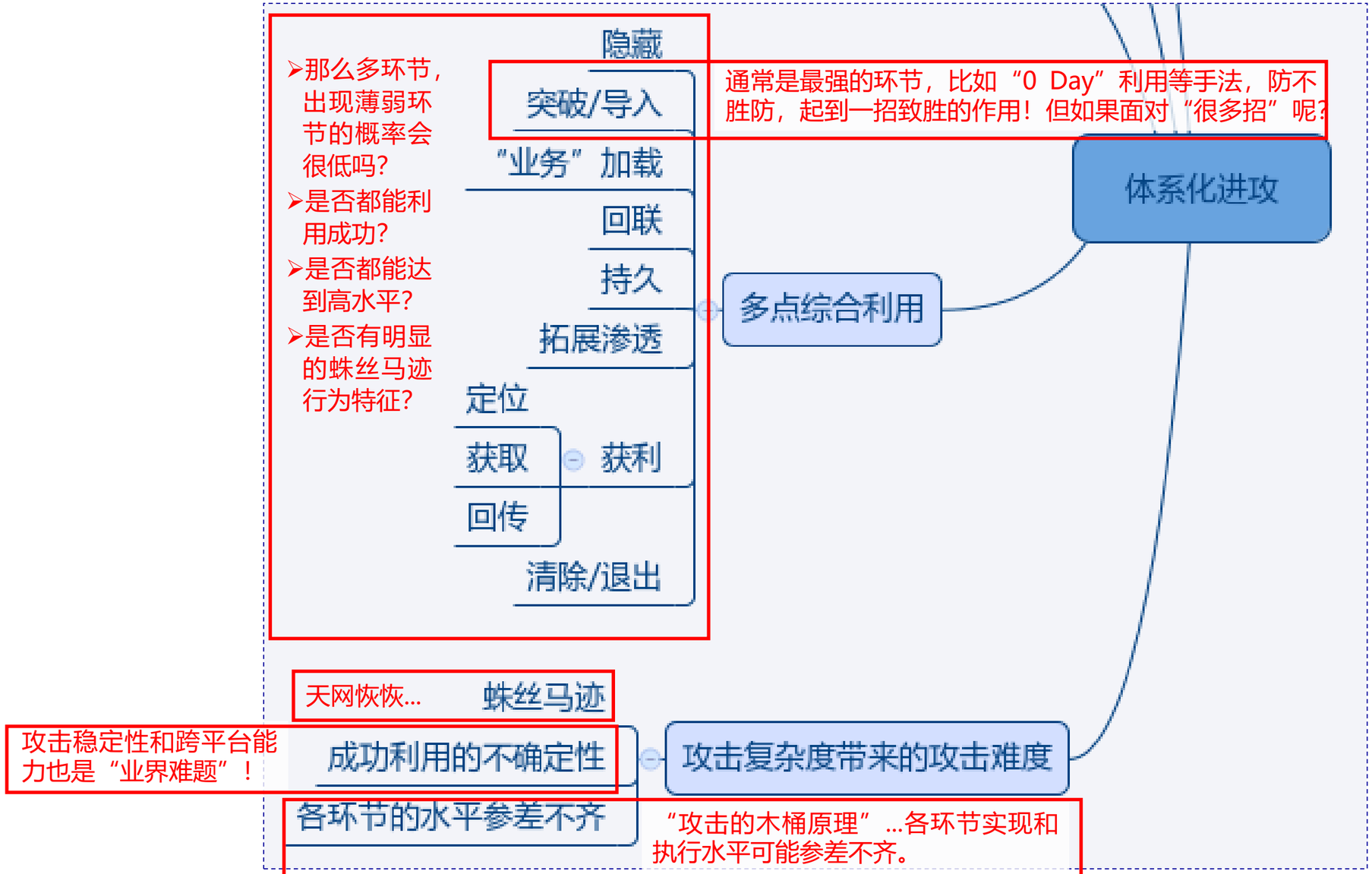


辩证看待挑战 — 体系化攻击并不“无懈可击”

的确，单一的防护机制在体系化进攻面前无能为力！但是如果假设我们拥有多样化的纵深防御能力，进攻行为的体系化形式，恰恰带来了更多的防御点——“进攻的薄弱环节”。



体系化攻击中的薄弱环节

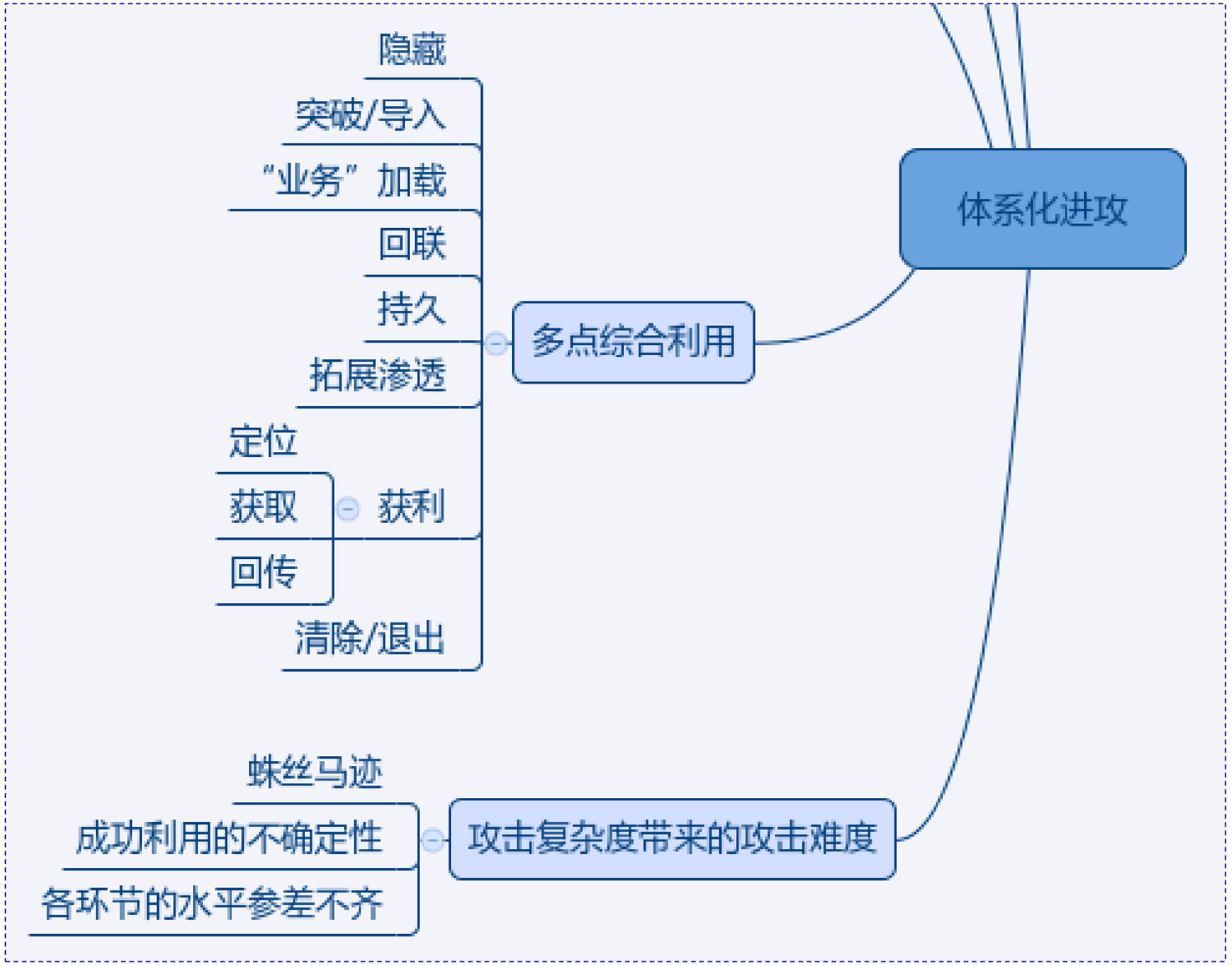


窃取信息攻击的薄弱环节 — 获利环节

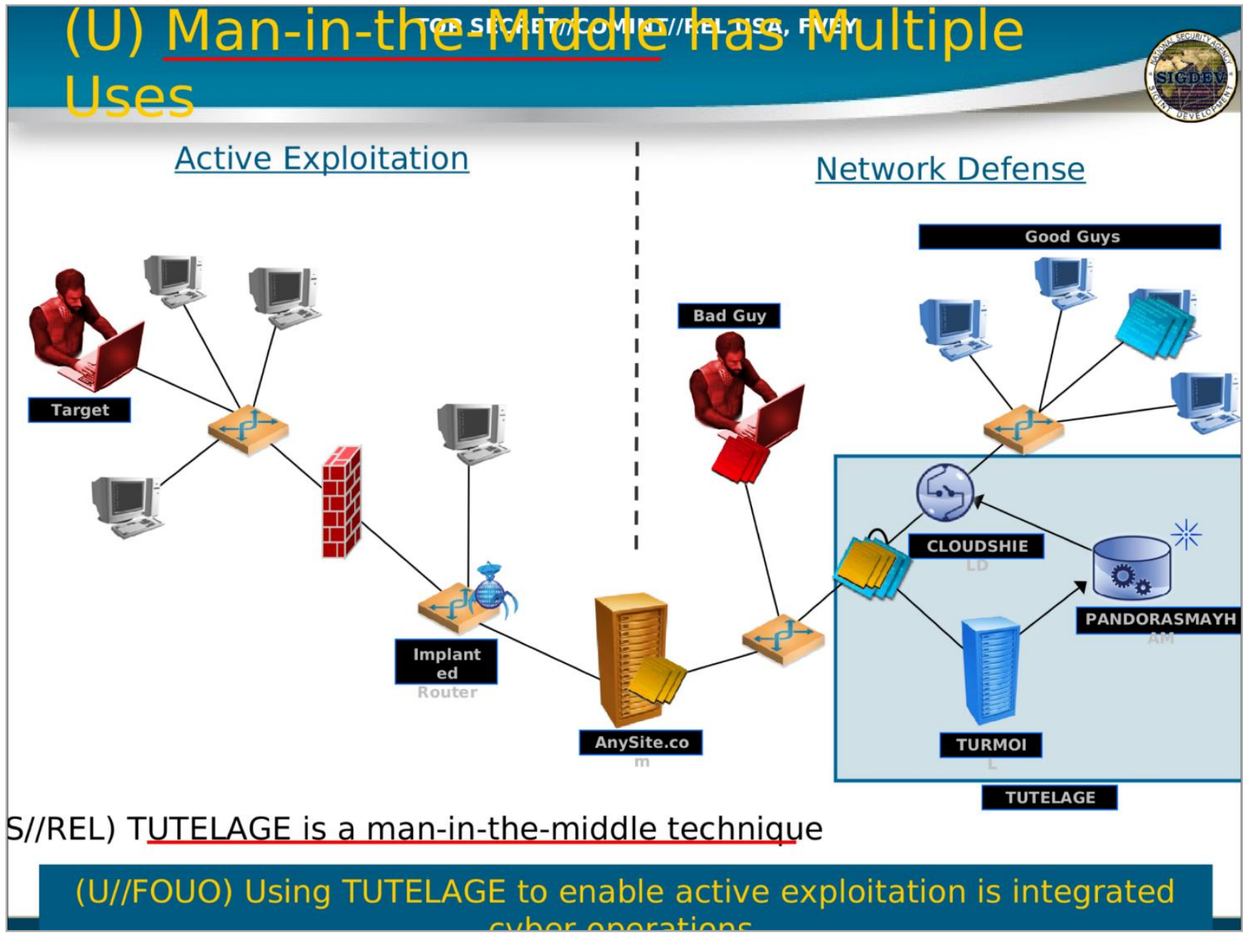
在分析窃取信息为目的的攻击并设计防御措施时，特别需要关注“窃取”这个获利环节。

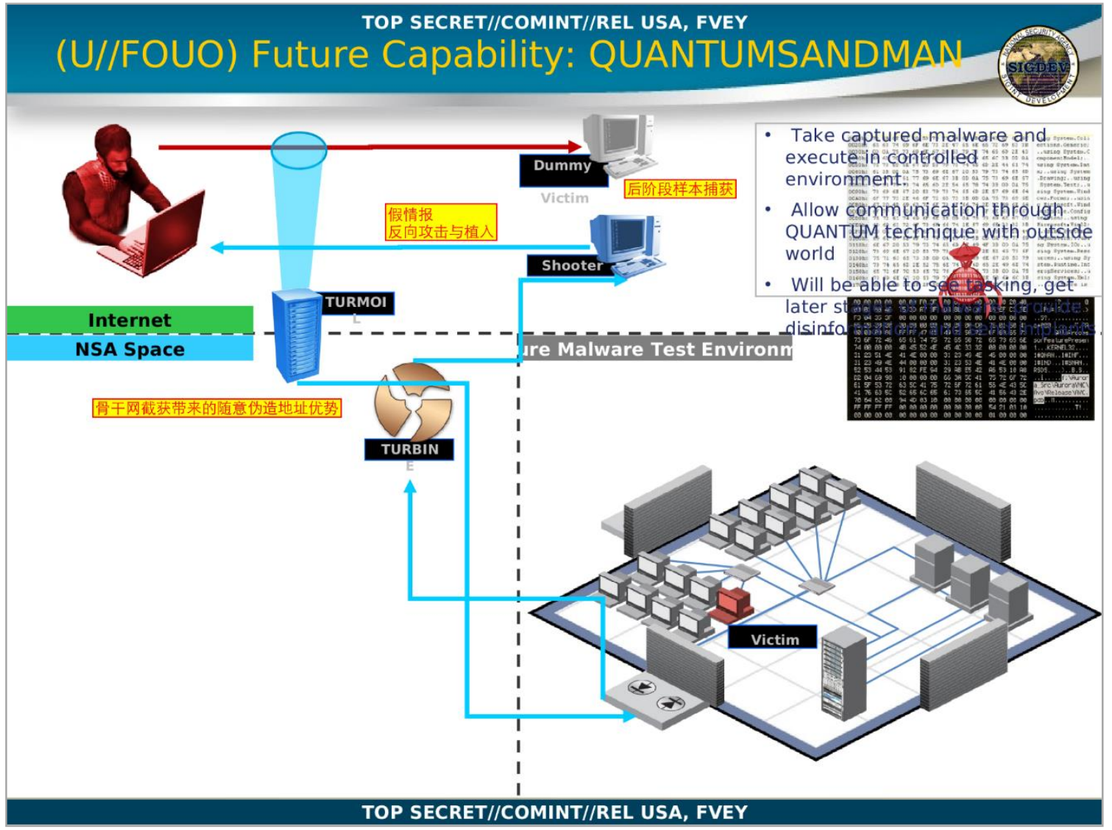
定位寻找有价值的信息，读取访问获得目标信息，以及各种渠道回传窃取的信息，从而实现攻击的目的。

如果没有获利环节，一次针对信息系统的攻击可能是没有效益的。另一方面，在整个攻击过程中，获利环节的隐匿性可能是最低的，而且由于攻击产业链的信任关系问题，其执行水平可能也是最差的！



直击“获利”环节 —— NSA的动态防御体系



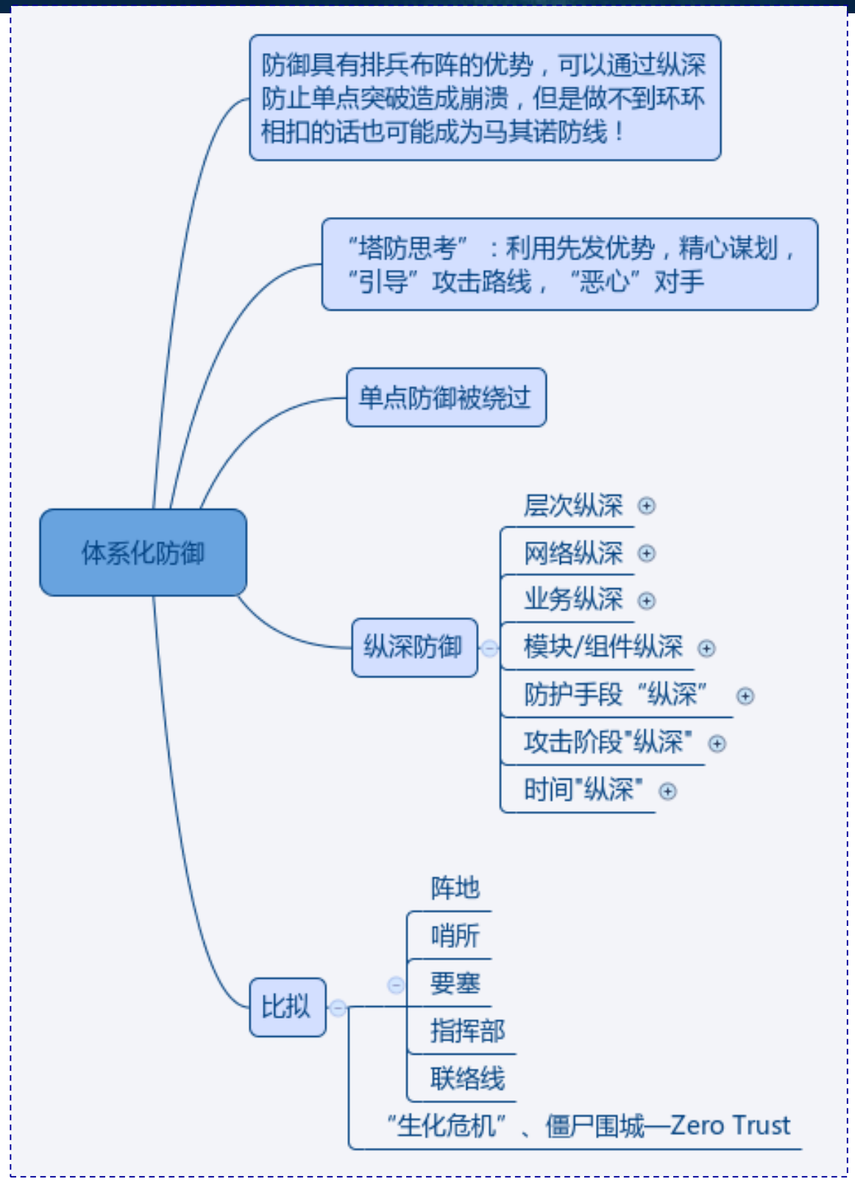


应对挑战 —— 构建纵深防御体系

攻击者已经联合起来了，已经形成分工合作的生态圈了，此时如果防守还是孤立、静态而且不成体系的，毋庸置疑成功者会是攻击者！

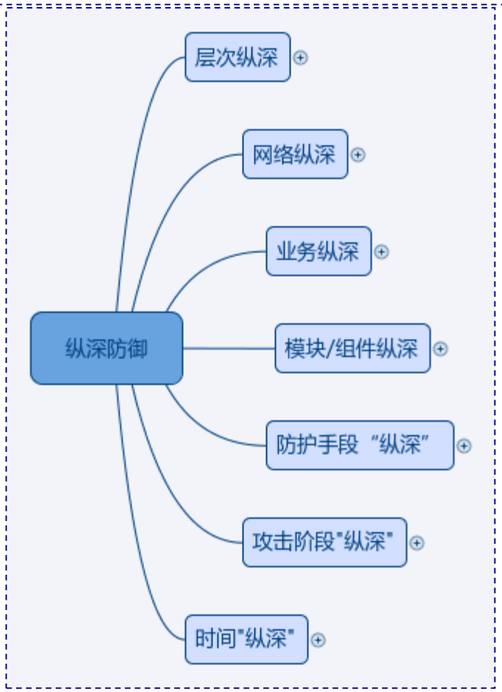
防守者只能利用好先发优势，布置好层层防线，综合利用多样化的手段，让进攻者在防守者布局的环境中“挣扎”！

传统的“纵深防御”原则，以及近些年火爆的“Moving Target Defence”，都强调体系化的防护！作为重要信息系统的防守方，我们**只能**抛弃“一招制敌”的幻想，“排兵布阵”构造**纵深防御**战线，体系化地与进攻者对抗。



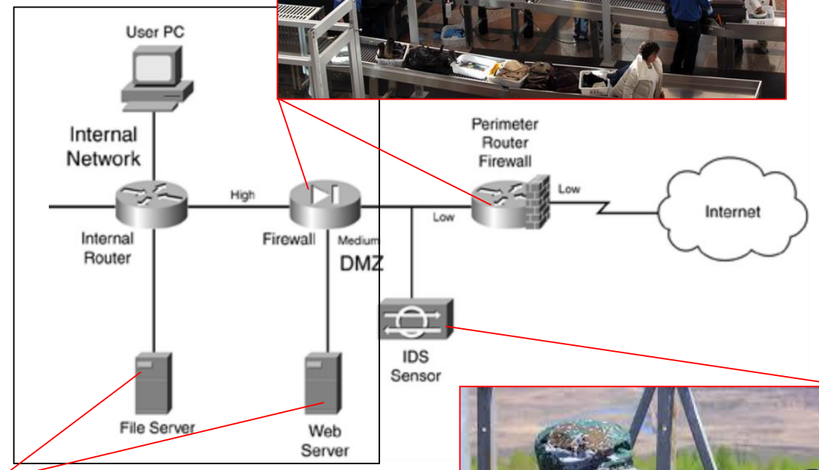
“纵深防御”是一种应该体现在信息安全防御体系设计各个方面的基本原则，而不是一种“可以独立堆叠形成的解决方案”。

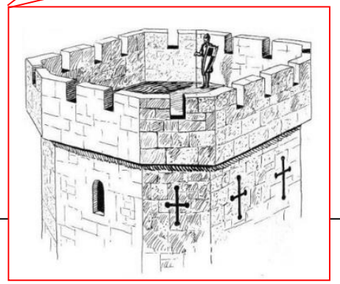
为了对抗体系化的攻击，防御体系的设计应用好“先发优势”，针对威胁行为模式，在各个方面防御手段的考虑纵深覆盖。



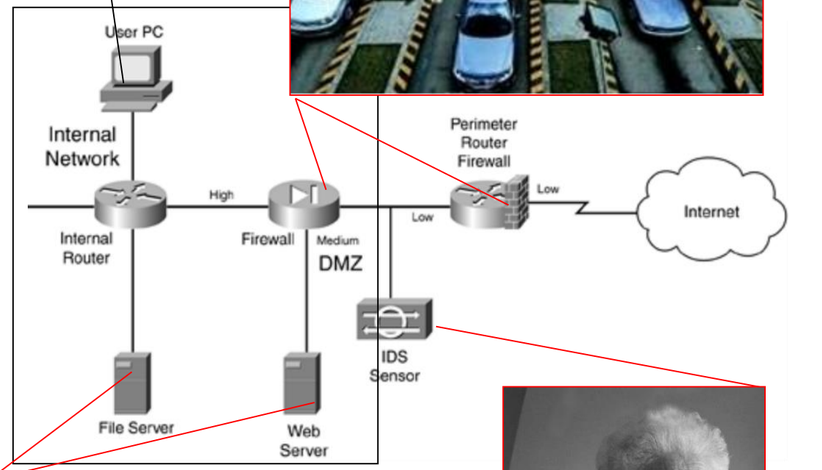
防御者需要直面“惨淡的现实”

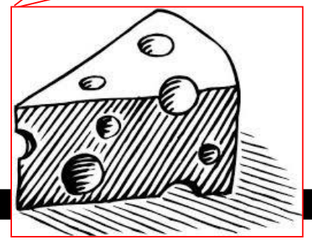
理想目标

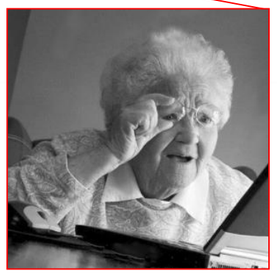




无奈现状



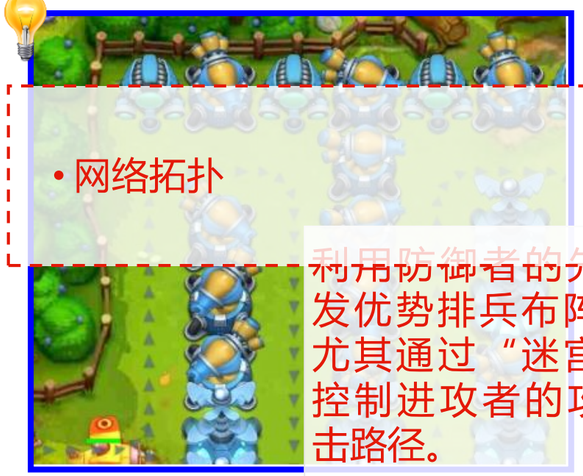




纵深防御的基本原则

1. Design for Failure
and nothing will really fail
"Everything fails, all the time"
Werner Vogels, CTO Amazon.com
Avoid single points of failure
Assume everything fails, and design backwards
Goal: Applications should continue to function even if the underlying physical hardware fails or is removed or replaced.

塔防游戏的启迪 —— 先发优势和纵深防御



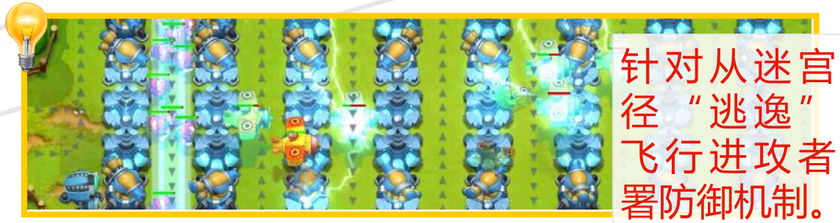
- 网络拓扑

利用防御者的先发优势排兵布阵，尤其通过“迷宫”控制进攻者的攻击路径。

形成多道防线，每一道防线都针对能通过前一道防线的进攻者特点，设计防御手段。

- 纵深防御
- Design for failure

地雷—避免损失的防御“后手”。



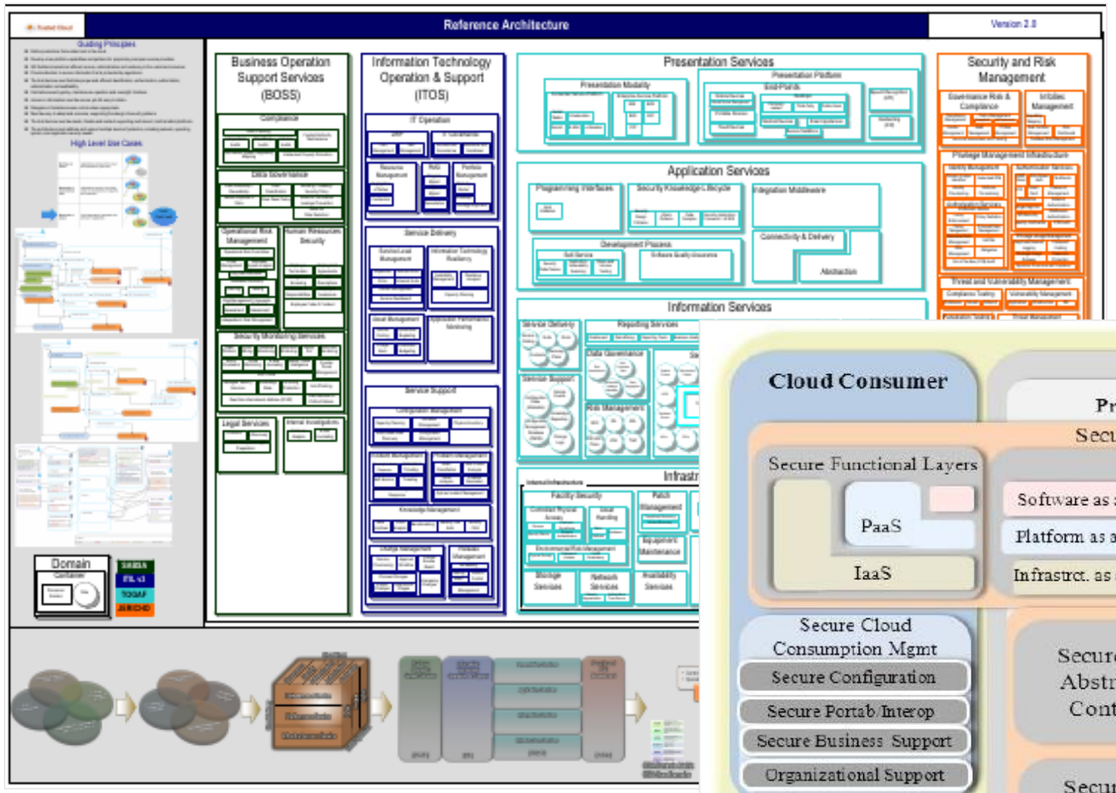
针对从迷宫路径“逃逸”的飞行进攻者部署防御机制。

Part

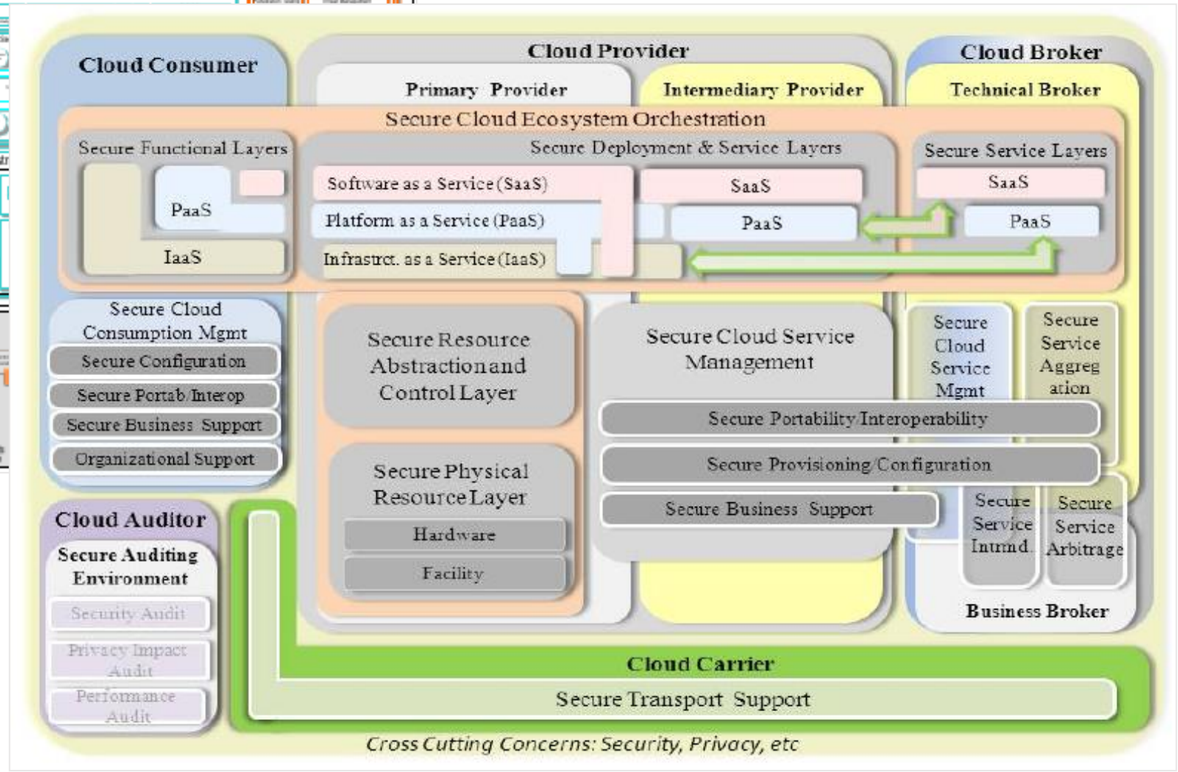
02

面向企业私有云安全的纵深防御实践

云计算安全框架的落地挑战



CSA和NIST都提出了较为完备的云安全框架，但是在如何实际的云计算环境中全面落实，一直是信息安全从业人员在云计算时代面对的挑战。



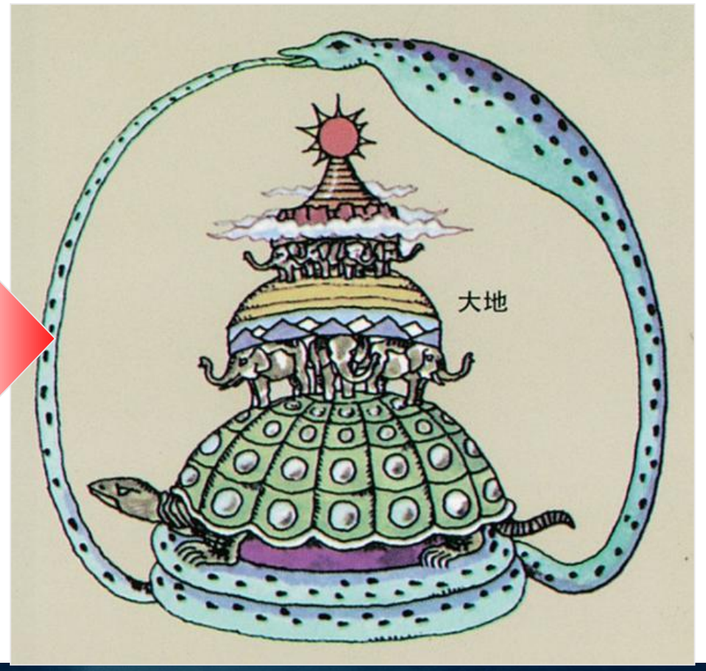
了解阵地 —— 从工程化实施视角看私有云

企业对私有云有更多的掌控，特别体现在**建设实施阶段**。与直接使用公有云“撒手不管”的“交付服务”视角不同，企业需要从**工程化实施视角**出发，关注如何规划、设计、建设、实施与推广使用私有云。

企业不仅需要从考虑如何使用云服务，还必须考虑并落实私有云建设和运维牵涉到的所有细节。

“交付服务”视角

工程化实施视角



了解阵地 —— 工程化实施视角下的“细节魔鬼”

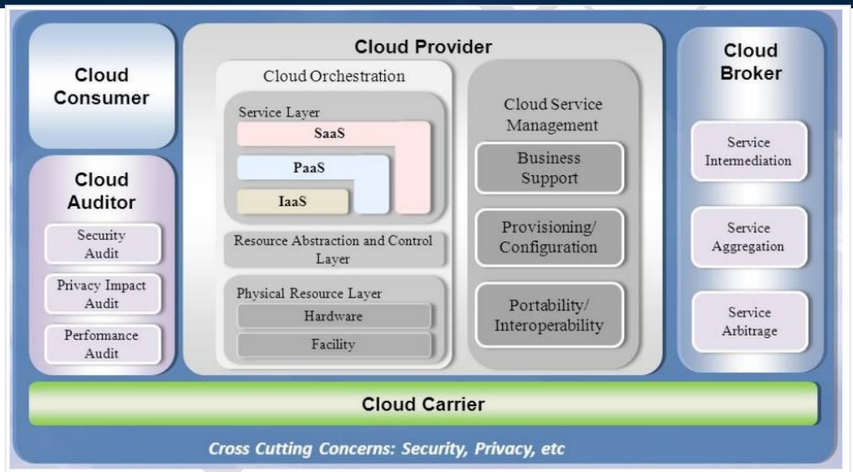
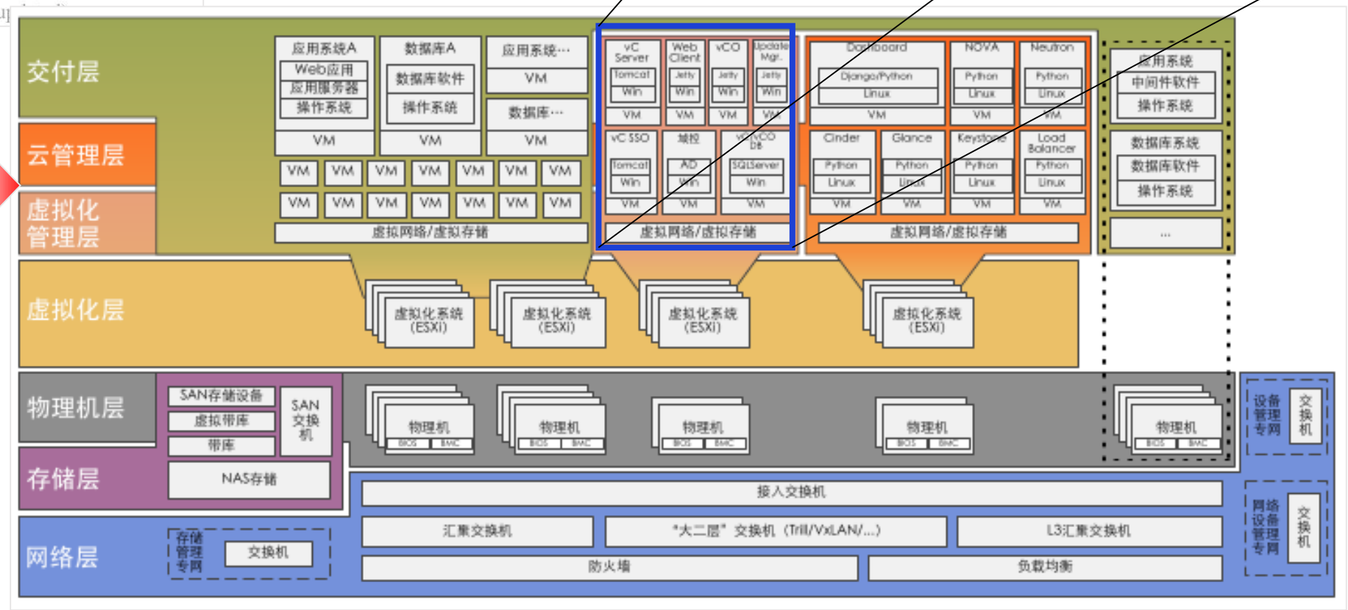


Figure 13: NIST Cloud Computing Reference Architecture (u

以NIST云计算参考架构为例，“交付服务”视角屏蔽了实施细节，具有较清晰的层次结构。

从更关注实施细节的工程化实施角度梳理典型云计算体系，可以发现云计算平台存在着多层次迭代的依赖关系，而且底层管理平台大量使用通用软件开发，**复杂度不亚于一套中小型信息系统**！



信息安全关注热点：

- 复杂度
- 迭代依赖关系
- 通用软件依赖

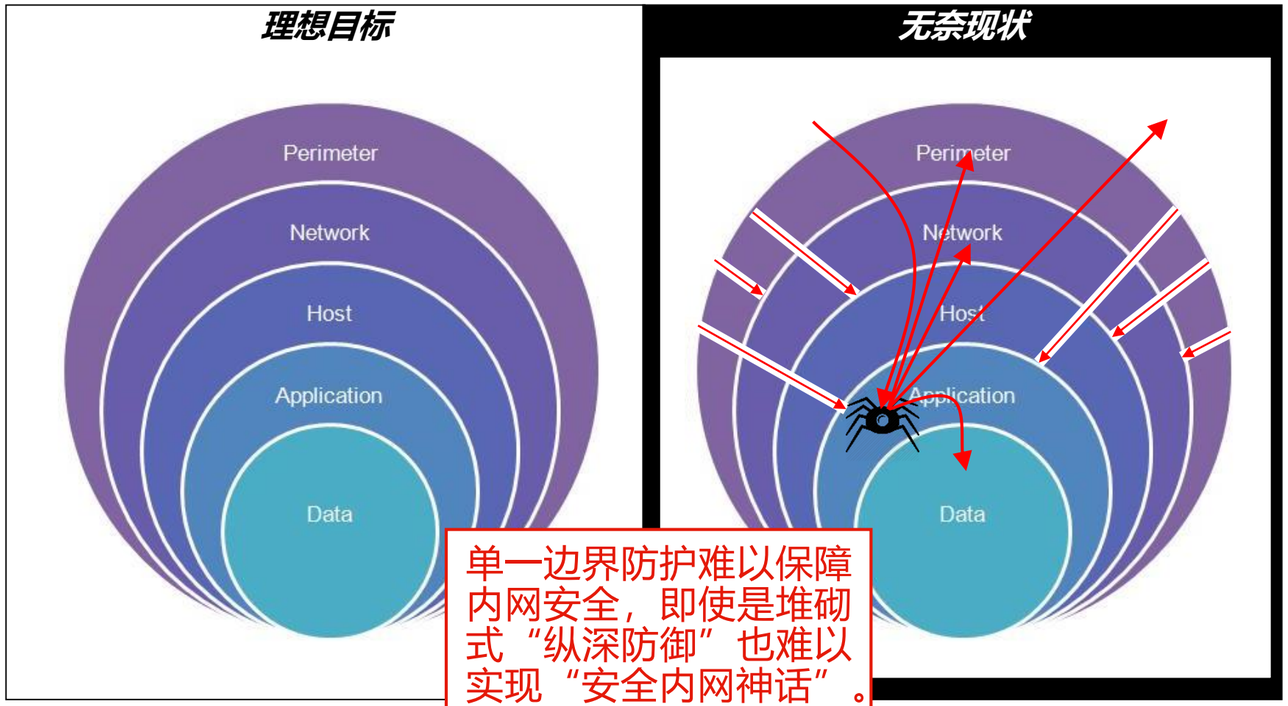
了解挑战 —— 边界防护的“无力感”

问：私有云在安全的内网里面啦，应该是安全的吧？

答：在当前复杂信息安全情况下，内网依然可能存在各种安全威胁。

问：如果觉得内网不够安全，加一层边界防护总可以了吧？

答：随着攻击逐渐上移至应用层并变得更复杂，传统的单一边界防护机制被绕过将会成为常态。



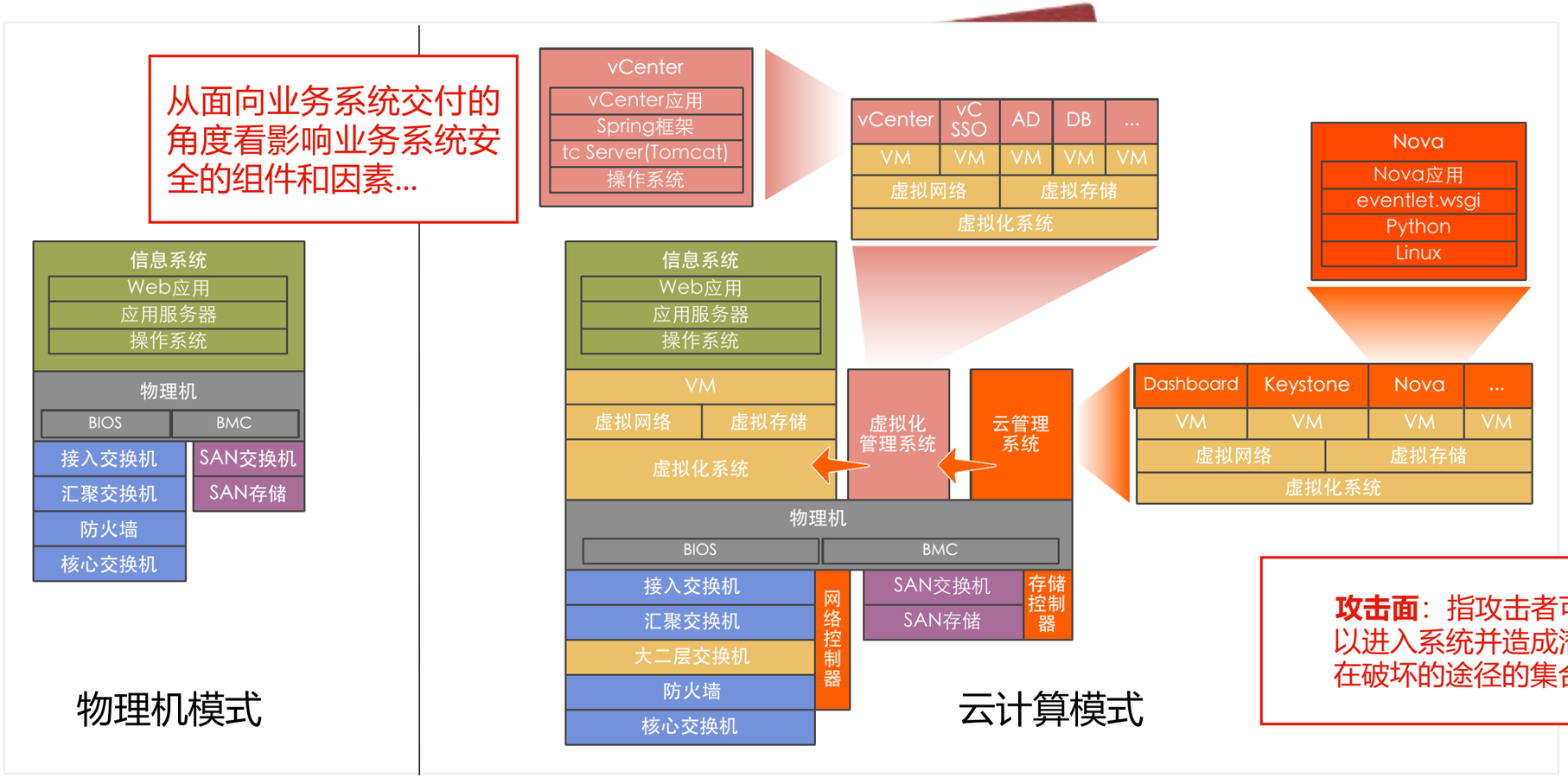
单一边界防护难以保障内网安全，即使是堆砌式“纵深防御”也难以实现“安全内网神话”。

私有云的安全防护压力并不会因为在内网而消失

了解挑战 —— “防御越来越难搞...”

问：据说云计算并不比传统物理机部署模式更不安全？

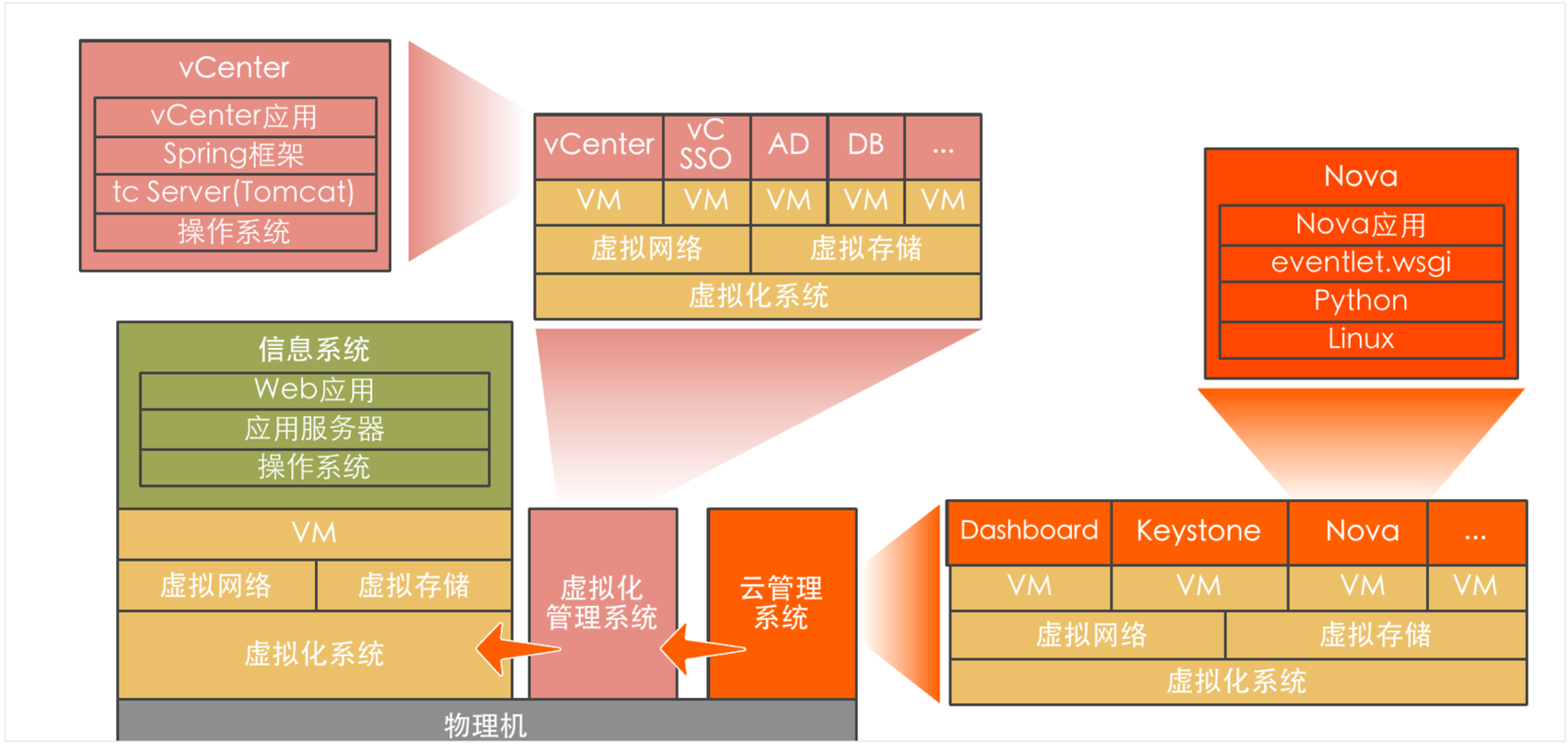
答：脱离实际情况比较哪个模式更安全没有意义。但从安全角度看，云计算确实引入了更多的系统层次，因此客观上具有**更多的攻击面**。



了解挑战 —— 还是那些熟悉的“宝贝疙瘩”

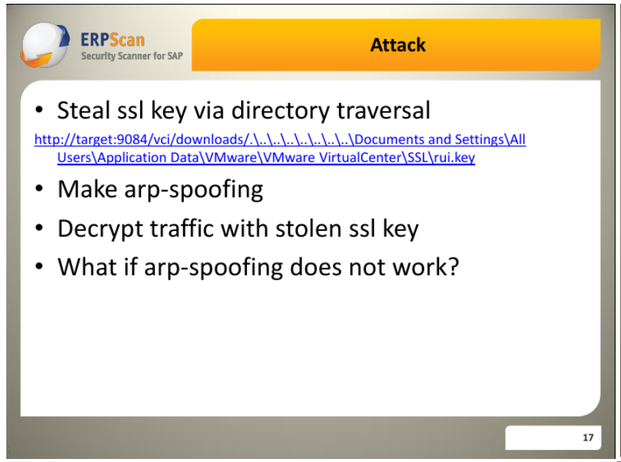
问：云计算技术改变了一切，传统的安全防护已经没有意义了吧？

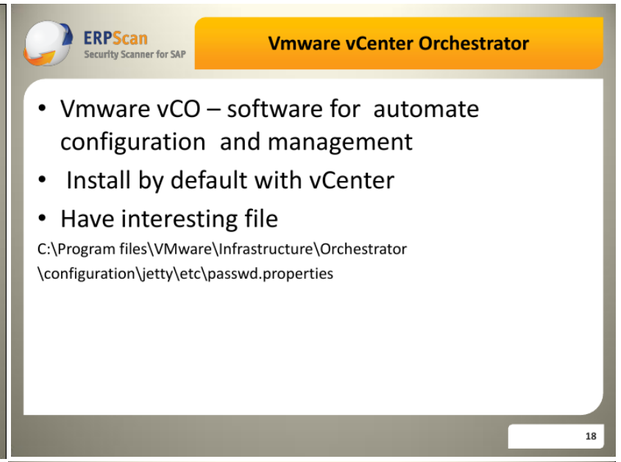
答：实际上云计算平台本身就是一个复杂的信息系统，特别是虚拟化管理与云管理系统采用通用软件和现有技术开发，最终也部署在传统硬件平台之上，依然受到传统软硬件技术生态圈的影响，因此传统攻击手段依然具有威胁性，还是**需要依靠传统防护手段作为私有云安全防护的基础**。

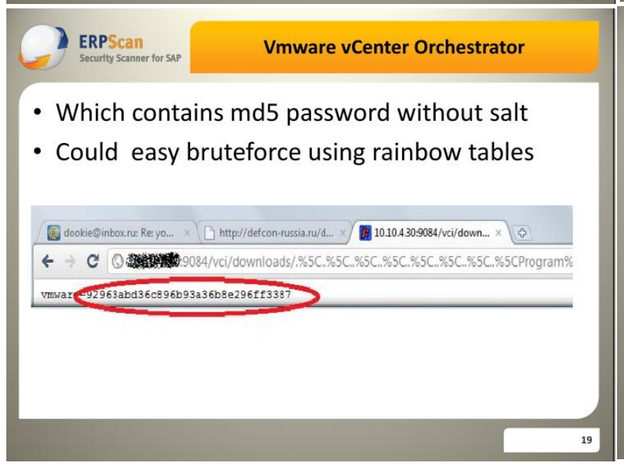


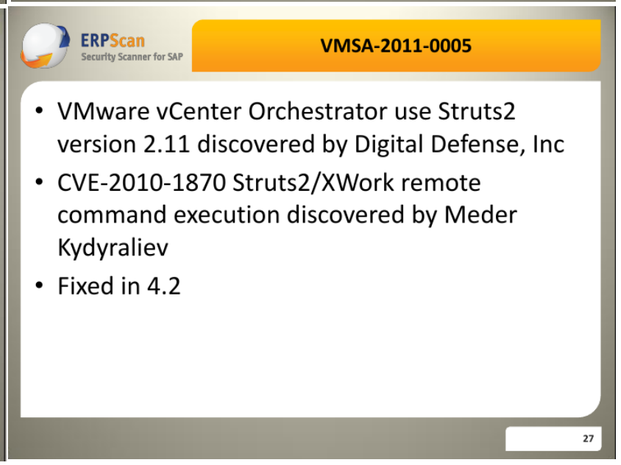
了解威胁 —— 对虚拟化管理平台的攻击

在Defcon 2012大会上，Alexander Minozhenko展示了利用Jetty的目录遍历漏洞对Vmware vCenter 4.1 Update 1版本的渗透，通过获得配置文件以及私钥文件，逐步控制虚拟化系统。而且该版本的vCO组件还存在Strusts2已知漏洞。





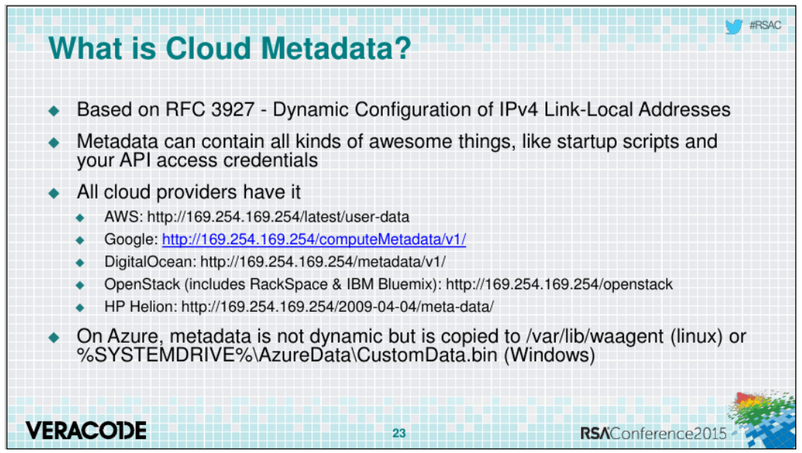


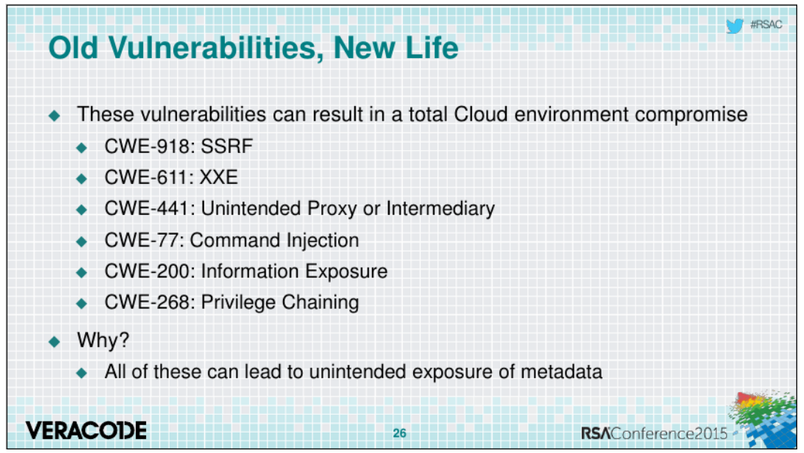


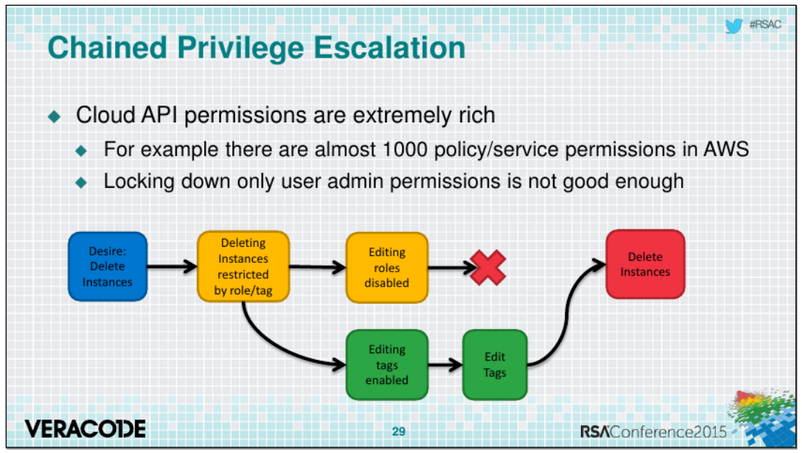
在云环境中，一旦虚拟化管理或云管理平台受攻击被控，租户系统只能被动地“默默承受”来自攻击者的恶意控制。

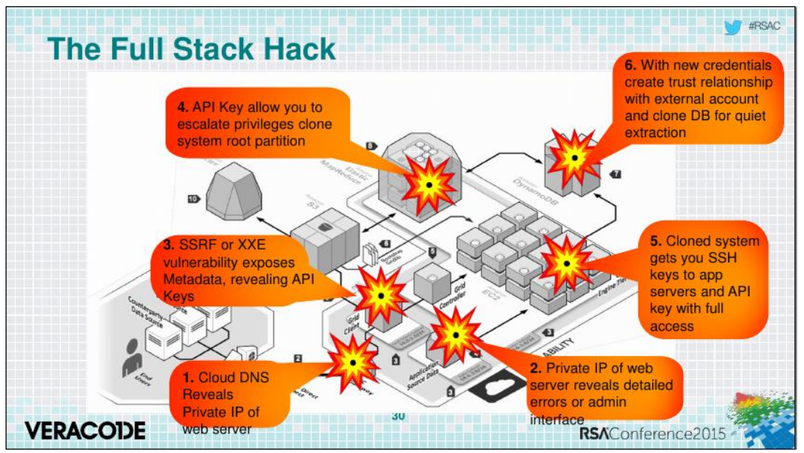
了解威胁 —— 传统手法造就云上全栈攻击

在Blackhat2015大会上，Erik Peterson提出了对云服务平台进行全栈攻击的概念，并提出通过一些常见的传统漏洞利用手段，可以获得对云计算服务元数据的控制，从而进一步完成各种攻击操作。







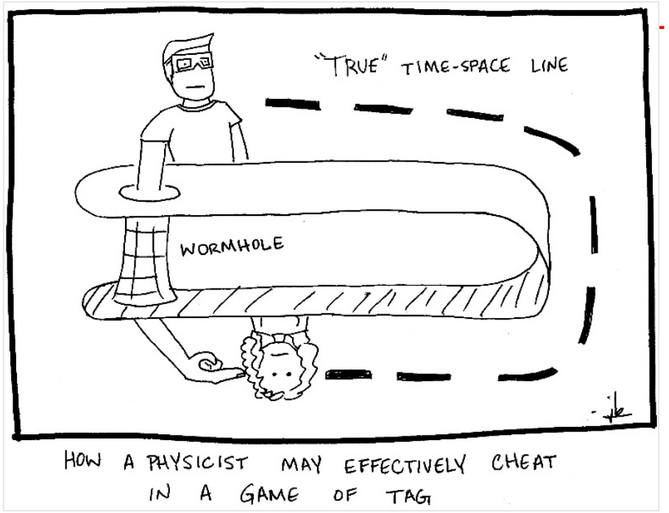


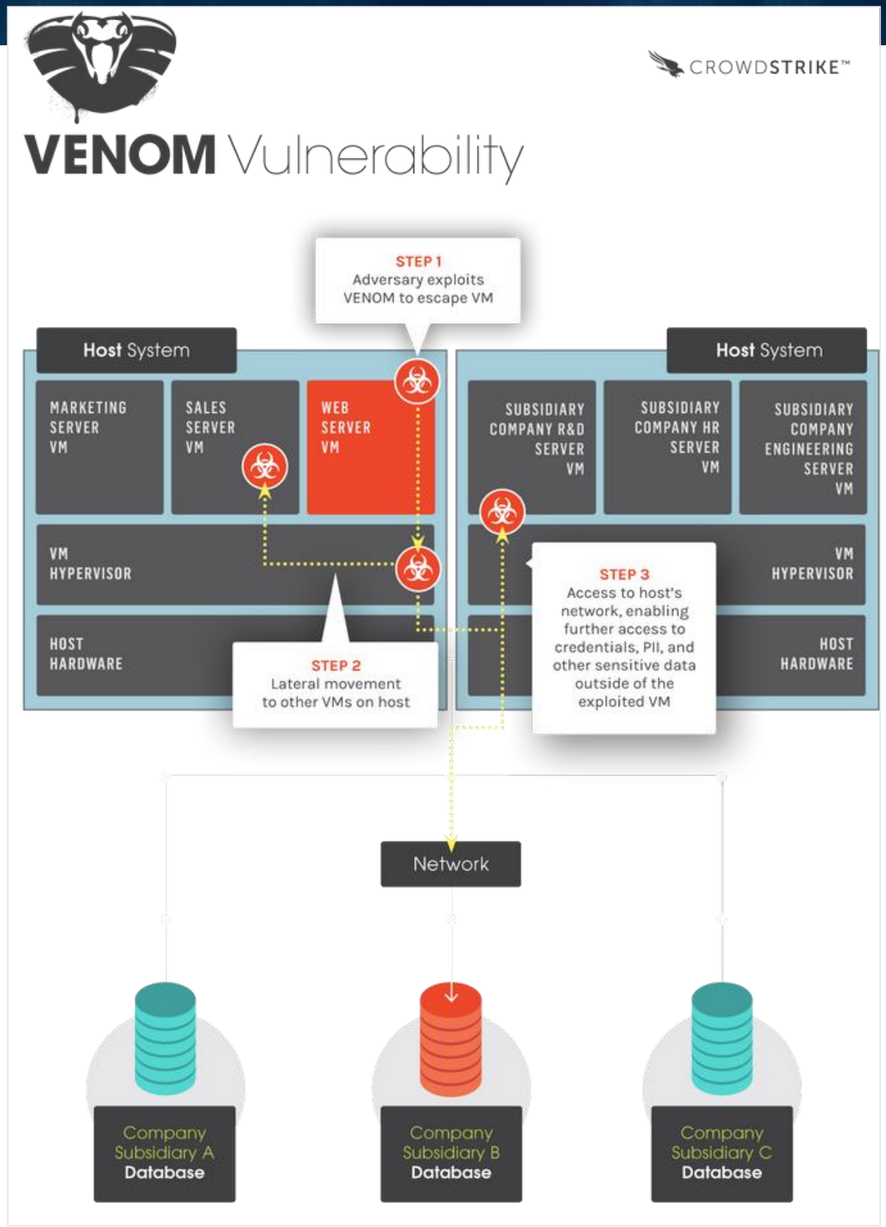
了解威胁 —— “所向披靡”的虚拟机逃逸

问：听说云安全的核心就是防止虚拟机逃逸，不过这玩意儿真的存在吗？

答：其实云安全不仅是防止虚拟机逃逸。姑且不说业界对于“数字军火商”们掌握逃逸漏洞的猜测，就在最近也有虚拟机逃逸漏洞公开“现身”了。

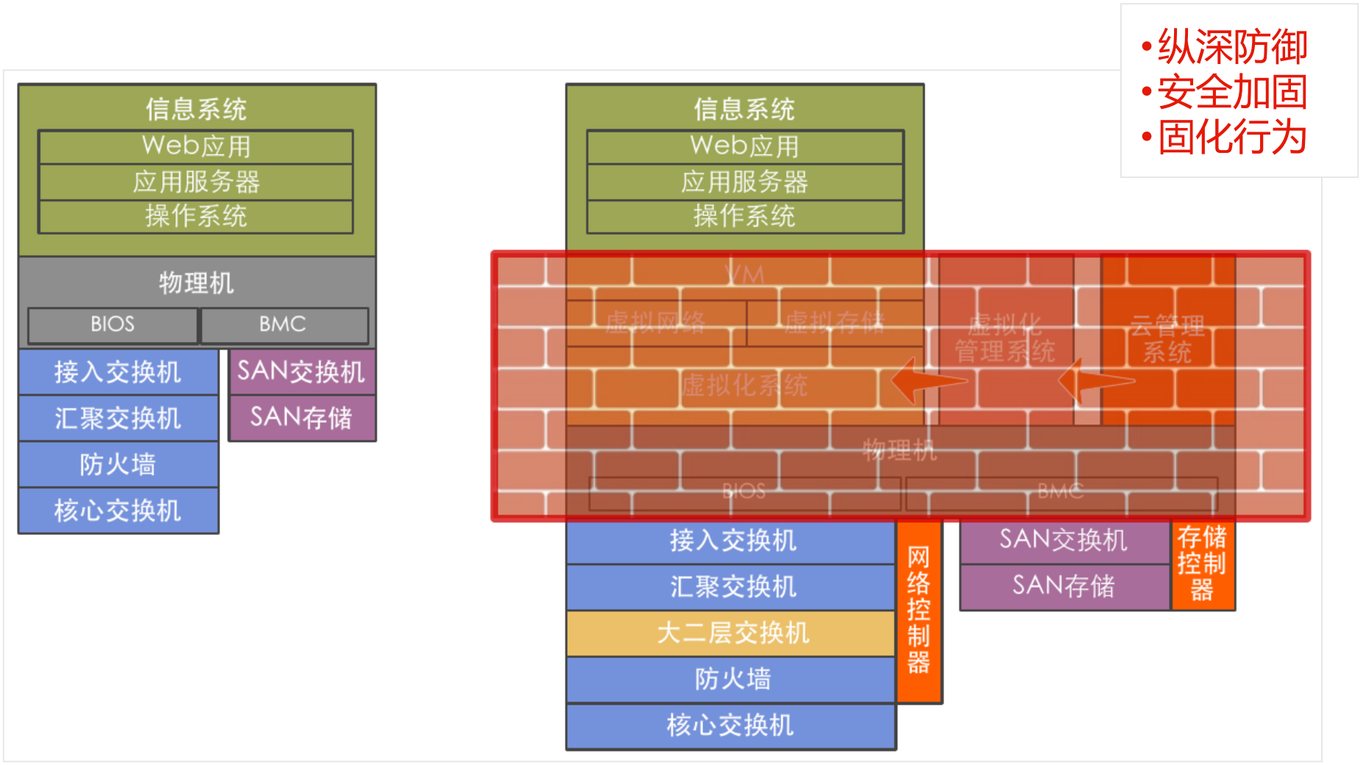
攻击者在完成逃逸后就如入无人之境了吗？有没有办法遏制攻击者呢？





防御目标定义

通过在工程化建设私有云的过程中实施基础安全防护措施，综合采用现有成熟的安全防护手段，面向主流的私有云技术体系，有效应对面向云计算平台底层的主要云安全威胁，为私有云平台的租户系统实现**不低于传统物理机模式的安全保障**。

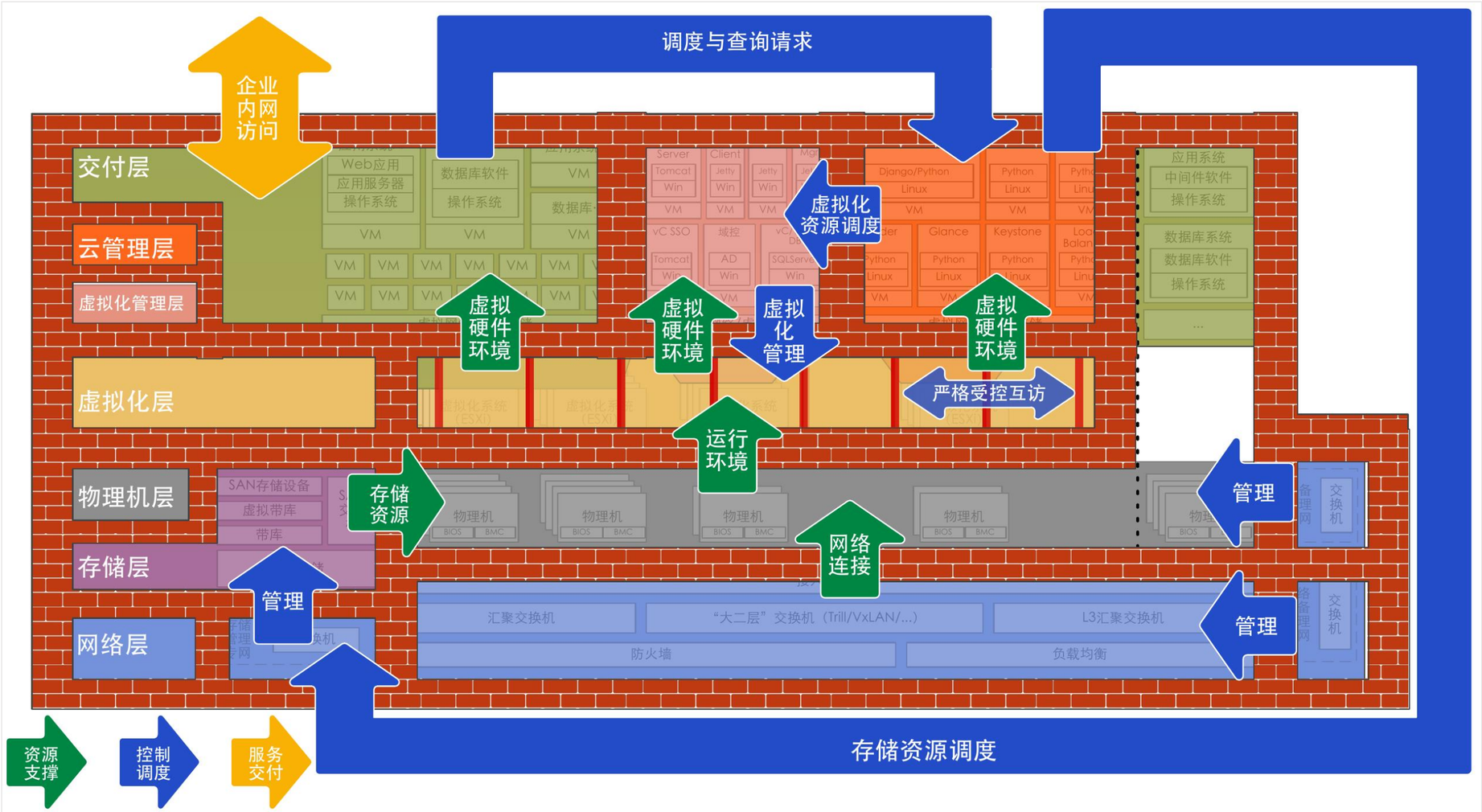


威胁领域	主要威胁内容
硬件	接入恶意设备或植入后门
	内部人员威胁
	供应链风险
固件/BIOS	固件级持久化或关闭设备的攻击
	对VM管理器/Hypervisor的攻击
	VM补丁更新不及时
	多租户场景下的跨虚拟机攻击
	对存储管理的攻击
操作系统	对操作系统镜像的攻击
	操作系统补丁更新不及时
	对操作系统的攻击以及植入恶意软件
中间件/基础服务	利用中间件配置问题的攻击
	监控不足以及人工干预
	对存在认证问题的服务的攻击
应用	数据控制问题；缺少保护的文件格式；缺少加密与访问控制
	SQL注入等针对设计和实现漏洞的攻击
	恶意应用程序（移动、桌面等）
网络基础设施	网络可恢复性问题
	网络隔离控制机制不成熟
	公有云服务将数据带入互联网甚至离境
	对路由器及其它网络设备的攻击
	网络设备供应链风险
	网络设备补丁更新管理不善

宏观先发 —— 采用分层控制体系构建纵深

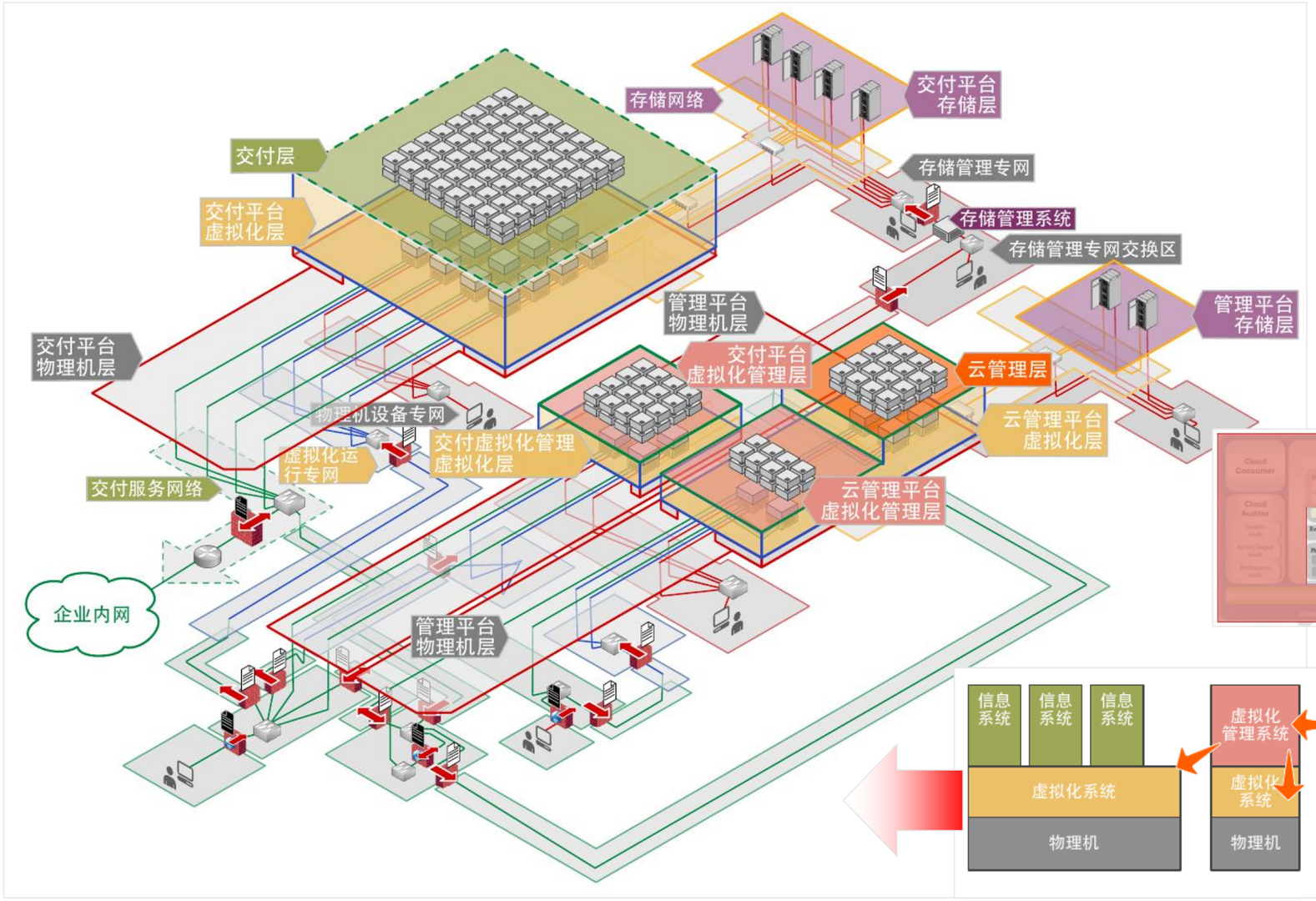
分层控制体系要点：

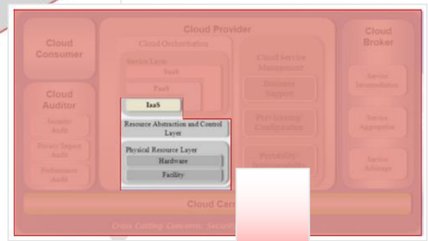
- 底层严格控制，保障交付层灵活应用；
- 各层功能内聚，降低运行依赖度；
- 分层自治，将云管理系统作为信任中心，接受统一管控；
- 多层纵深控制，固化底层行为；
- 收缩硬件管理访问接口，形成管理专网。

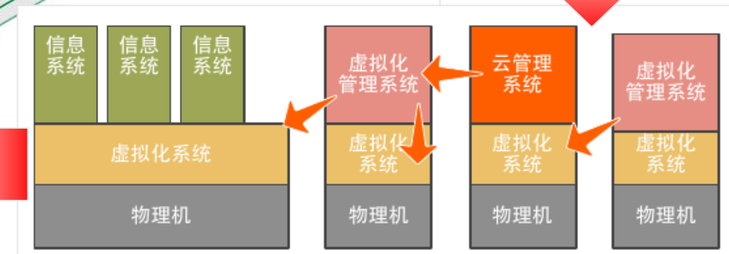


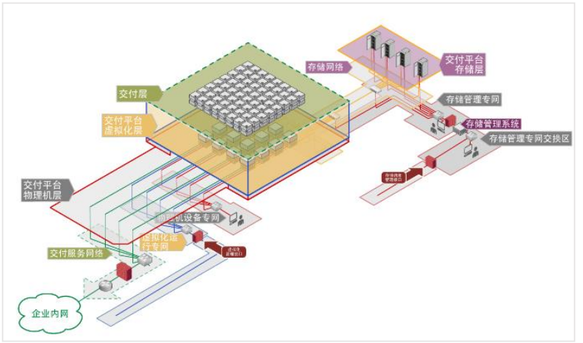
可承载纵深防御体系的分层自治部署架构

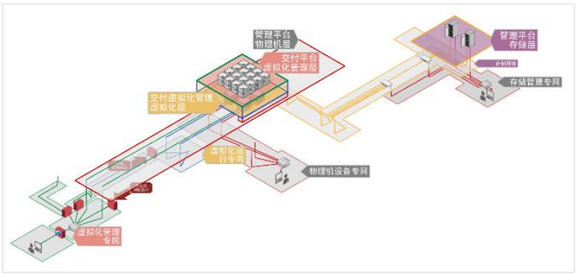
基于分层控制模型，以3D模型的形式梳理并设计私有云部署架构。

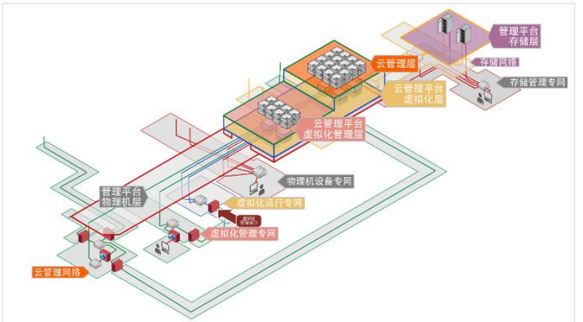






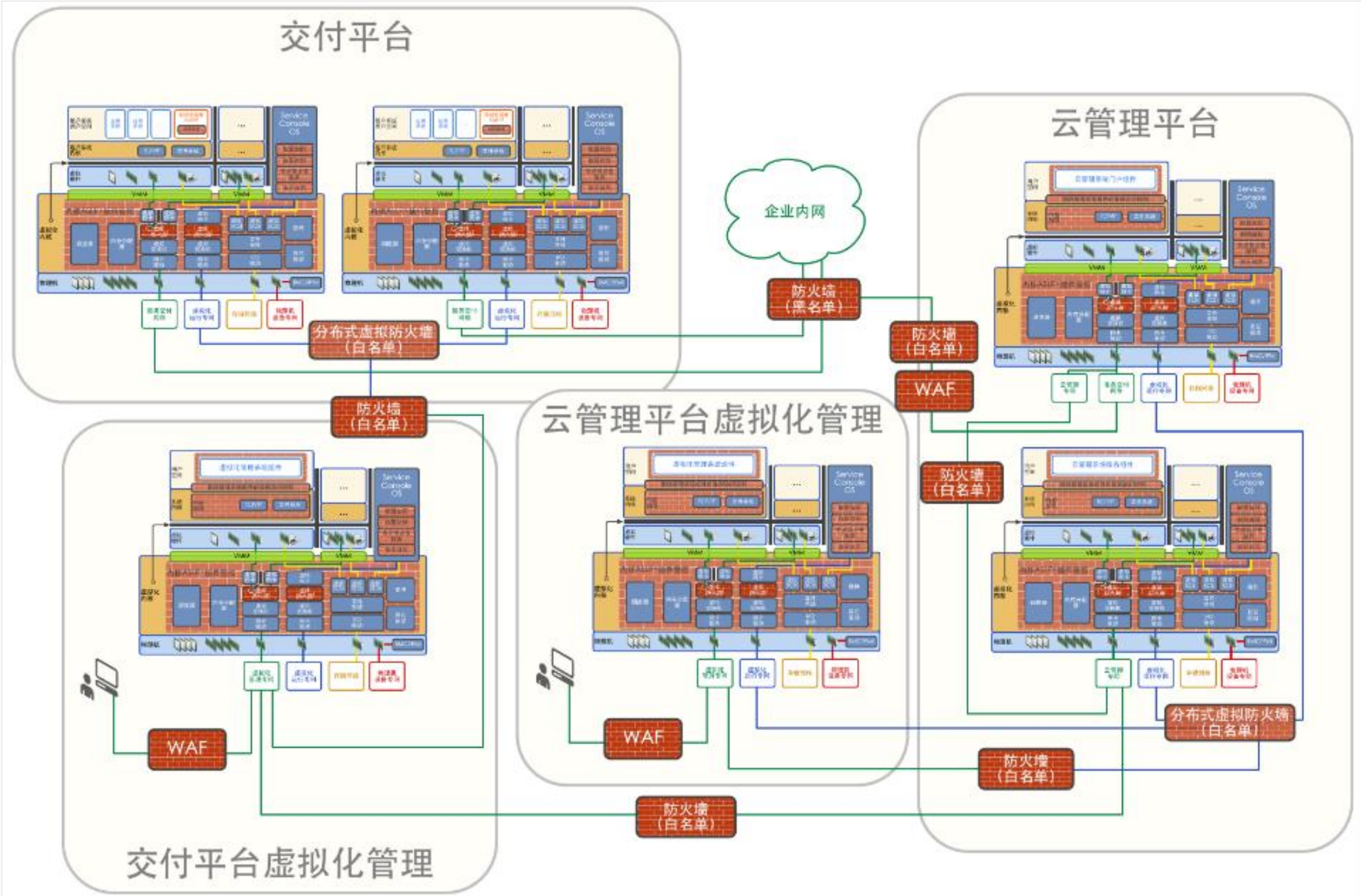




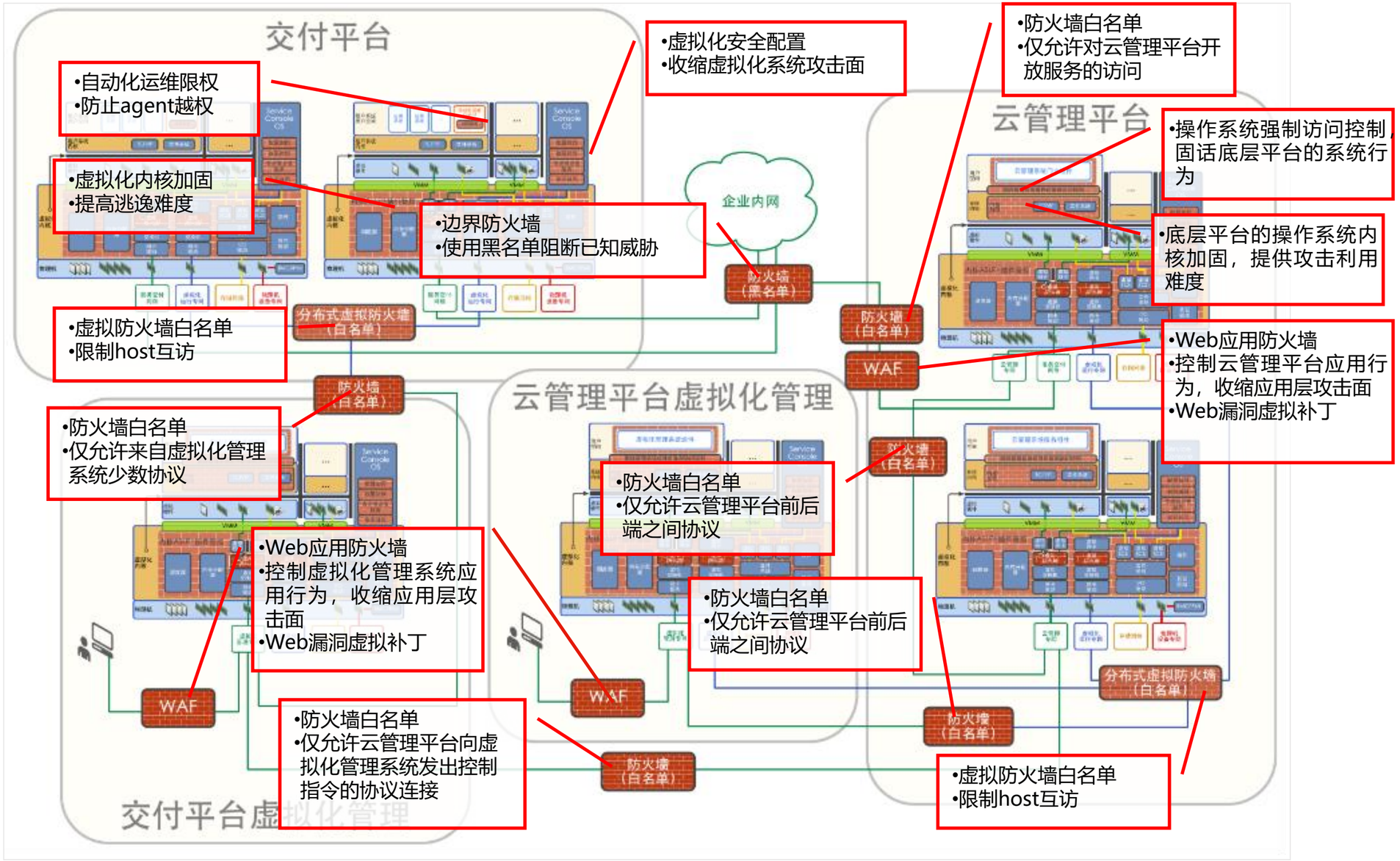


中观先发优势 —— 多层次的排兵布阵

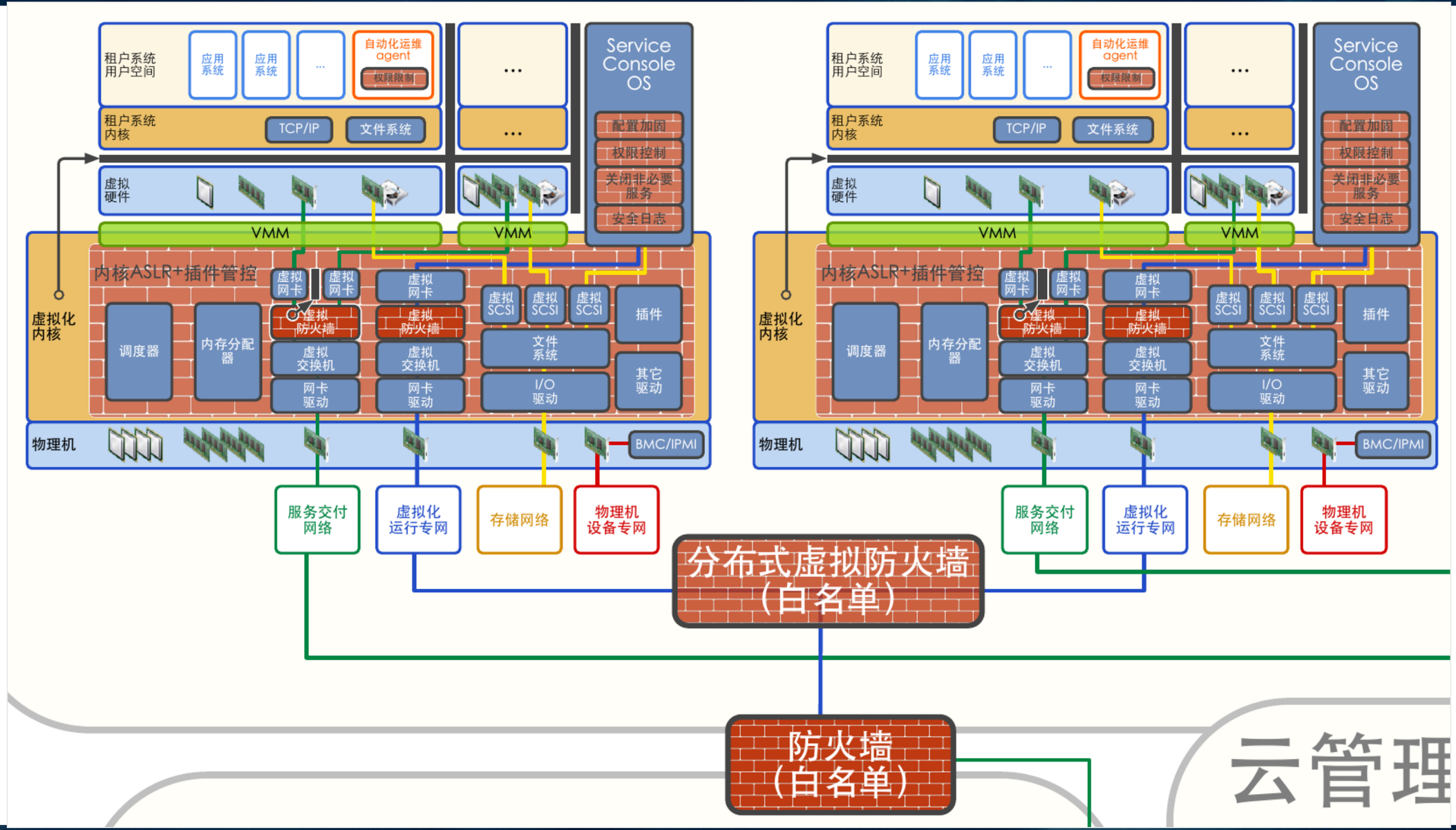
- 结合私有云技术体系开展纵深防御，调整网络拓扑与部署架构；
- 依托网络纵深，设计多道防线；
- 网络防线采用白名单策略，收缩攻击面并固化底层行为；
- 加固虚拟化系统，防范虚拟机逃逸，并依托网络纵深防线控制逃逸后攻击路径；
- 严格加固底层平台的应用软件和操作系统，防止跨维度攻击；
- 借鉴移动目标防御(MTD)思路，注重逐层收缩攻击面，防范“一洞打穿”。

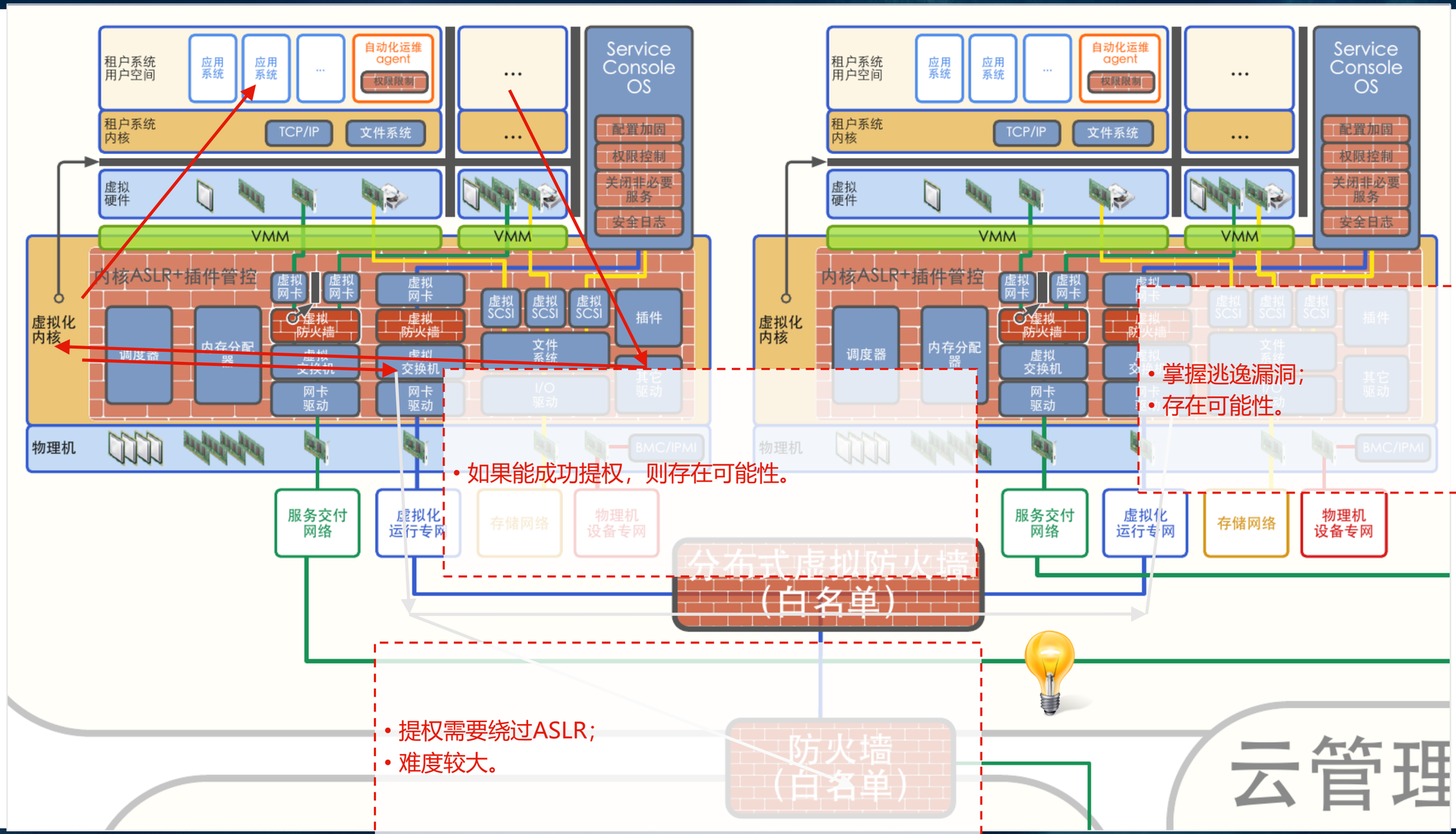


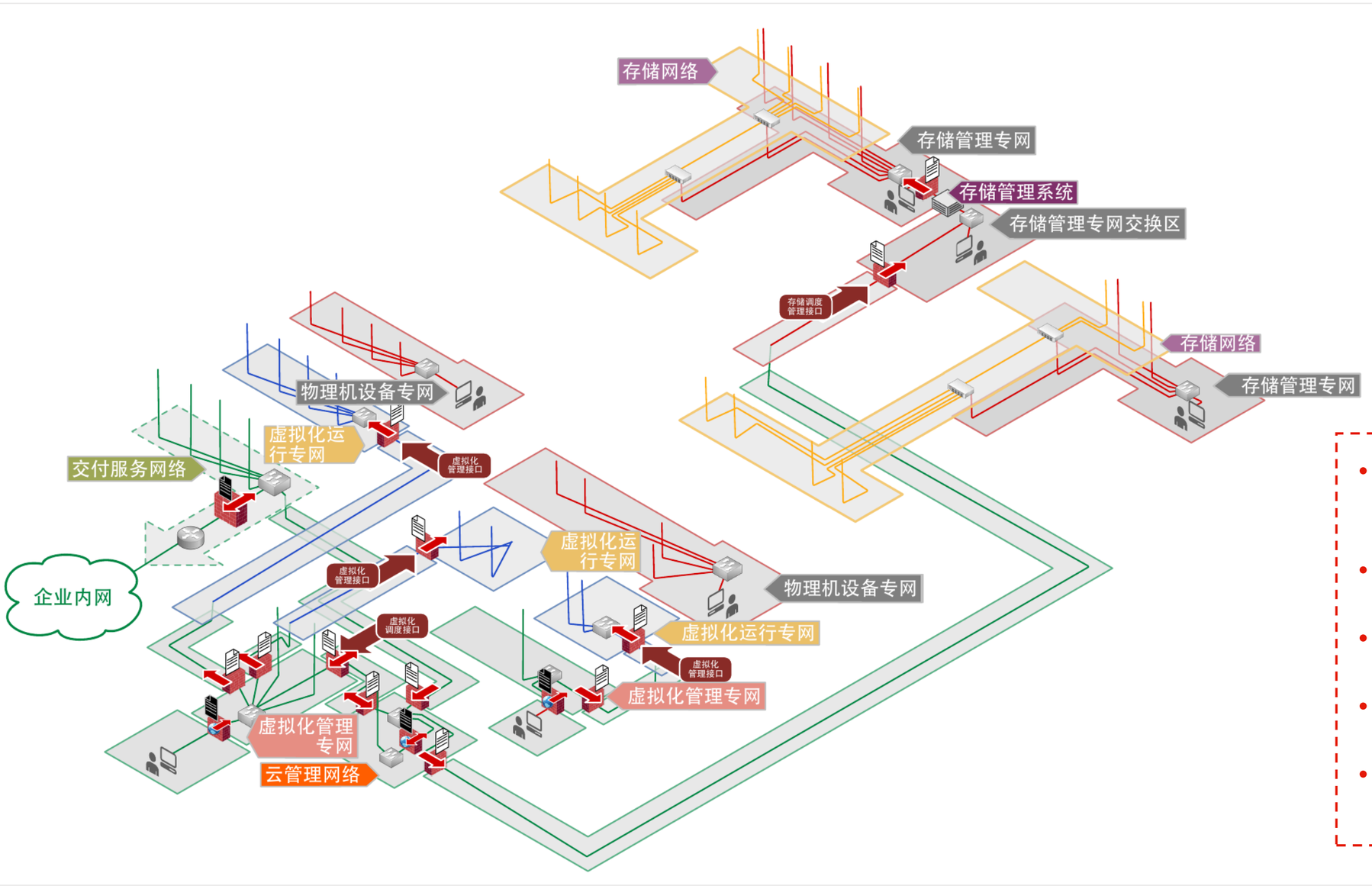
环环相扣 —— 多层次有机结合的防御手段



防御阵地一角

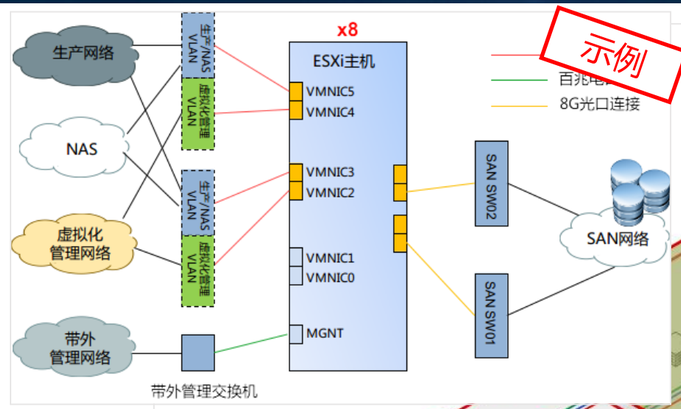

租户系统
用户空间
应用系统
应用系统
...
自动化运维agent
权限限制
租户系统
内核
TCP/IP
文件系统
虚拟硬件
VMM
VMM
Service Console OS
配置加固
权限控制
关闭非必要服务
安全日志
虚拟化内核
内核ASLR+插件管控
虚拟网卡
虚拟网卡
虚拟网卡
虚拟防火墙
虚拟防火墙
虚拟SCSI
虚拟SCSI
虚拟SCSI
插件
调度器
内存分配器
虚拟交换机
虚拟交换机
文件系统
网卡驱动
网卡驱动
I/O驱动
其它驱动
物理机
BMC/IPMI
服务交付网络
虚拟化运行专网
存储网络
物理机设备专网
分布式虚拟防火墙（白名单）
防火墙（白名单）
云管理

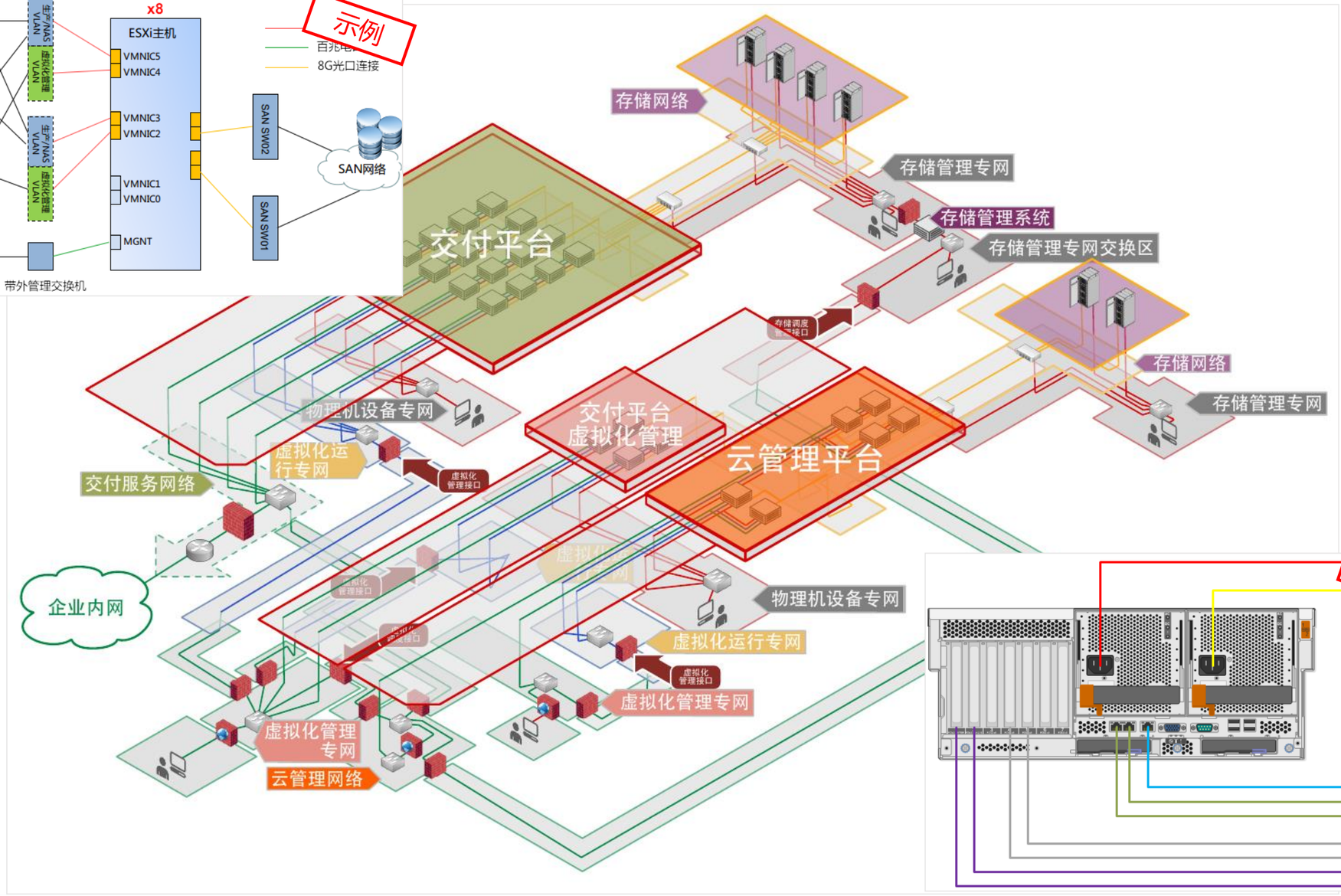

租户系统
用户空间
应用系统
应用系统
...
自动化运维agent
权限限制
租户系统
内核
TCP/IP
文件系统
虚拟硬件
VMM
Service Console OS
配置加固
权限控制
关闭非必要服务
安全日志
虚拟化内核
内核ASLR+插件管控
虚拟网卡
虚拟防火墙
虚拟交换机
网卡驱动
虚拟SCSI
文件系统
插件
调度器
内存分配器
I/O驱动
其它驱动
物理机
BMC/IPMI
服务交付网络
虚拟化运行专网
存储网络
物理机设备专网
分布式虚拟防火墙
（白名单）
防火墙
（白名单）
• 如果能成功提权，则存在可能性。
• 掌握逃逸漏洞；
• 存在可能性。
• 提权需要绕过ASLR；
• 难度较大。
云管理

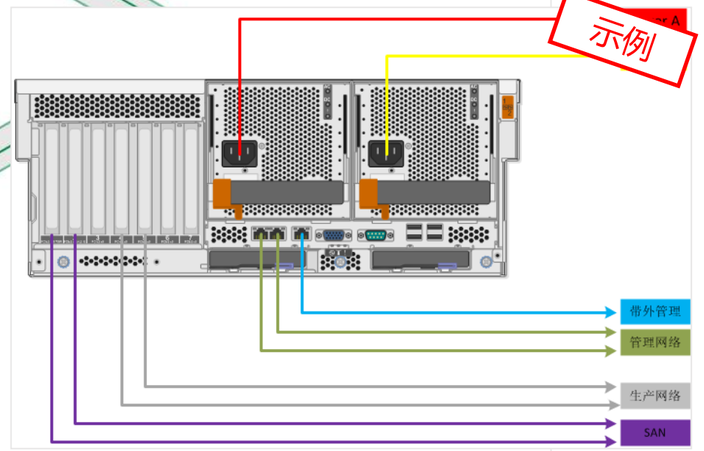


- 由于“功能内聚”的设计原则，维持各层运行的组件可被纳入最小化的安全区域；
- 遵循“分层自治”原则，收缩区域接口，使用白名单固化行为；
- 使用WAF应对面向底层组件应用系统的威胁；
- 将缺少安全防护但风险极大的存储与设备管理接口纳入专网；
- 交付平台的虚拟化和存储管理层具有最大的访问路径防御纵深。

纵深防御落地 —— 部署实施


生产网络
NAS
虚拟化管理网络
带外管理网络
生产/NAS VLAN
虚拟化管理 VLAN
x8
ESXi主机
VMNIC5
VMNIC4
VMNIC3
VMNIC2
VMNIC1
VMNIC0
MGNT
SAN SW02
SAN SW01
SAN网络
带外管理交换机
示例
8G光口连接


存储网络
存储管理专网
存储管理系统
存储管理专网交换区
存储网络
存储管理专网
交付平台
交付平台
虚拟化管理
云管理平台
物理机设备专网
虚拟化运行专网
虚拟化管理接口
交付服务网络
企业内网
物理机设备专网
虚拟化运行专网
虚拟化管理接口
虚拟化管理专网
虚拟化管理专网
云管理网络


示例
带外管理
管理网络
生产网络
SAN

系统安全防御机制落地的困惑...

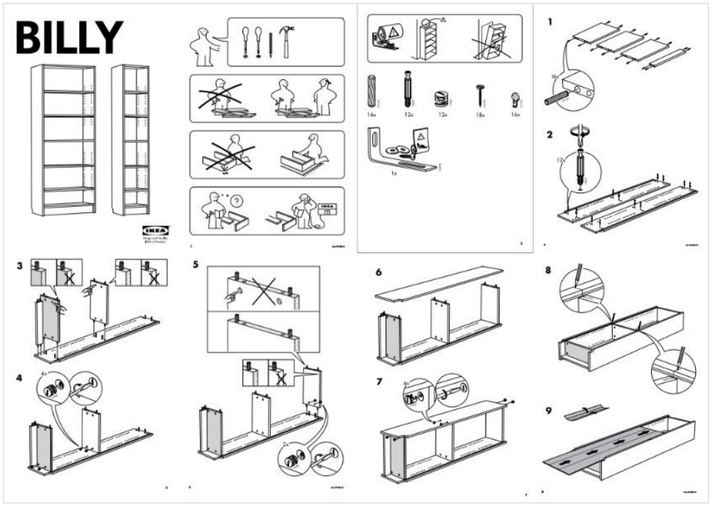
根据实践经验，企业在建设私有云过程中，常常在如果落实安全防护措施方面显得不知所措，经常会出现甲方提出高阶需求后无法细化，最后出现“放任”实施团队根据经验“想到哪、做到哪”的窘况。

私有云 功能建设	规划	概要 设计	详细 设计	建设 实施	推广 上线
私有云 安全建设	（发呆...）	罗列 需求点	收罗 安全手册		“扫描” 验证

参考
手册

以控制为中心

VS



参考
手册

以人为中心

纵深防御落地 —— 防御拓展至建设阶段

通过私有云建设实践，发现虽然各虚拟化系统厂家都提供了丰富的安全指南，但在实际工程化过程中，实施工程师面对具体配置项常常会无所适从或出现遗漏。因此需要根据实际私有云的设计方案，对安全指南的加固措施集进行梳理裁剪，形成面向工程化实施的加固方案。

类别	名称	加固要求	风险等级	处置措施	安全判定依据	加固操作参考
访问控制	控制外部访问	对于ESXi运行的服务均均通过ESXi防火墙配置允许访问的主机	1,2,3	1）按照ESXi防火墙策略表配置，关闭不必要的端口 2）对于开放的端口，仅允许特定的主机访问，如vCenter/堡垒区主机等	通过vSphere web client或vSphere Client连接到vCenter，打开一个ESXi—>管理—>安全配置文件，查看防火墙部分的配置	通过vSphere web client或vSphere Client连接到vCenter，选择一个ESXi—>管理—>安全配置文件，查看已经启动的服务；在FireWall部分点击编辑，针对启动的服务设定允许访问的外部主机
其他	配置可信NTP	配置				
日志	配置持久保存成的日志存储位置	为				
访问控制	禁用DCUI	禁用				
访问控制	禁用ESXi Shell	禁用				
访问控制	禁用mob	禁用				
访问控制	禁用ssh	禁止				
	配置Host Profile	采用				

ESXi (15项)

类别	名称	加固要求	风险等级	处置措施	安全判定依据	加固操作参考
操作系统	加固vCenter、SSO、Update Manager所在的windows操作系统	使用受支持的操作系统版本，确保操作系统的升级包和补丁的更新满足企业一般安全策略的要	1,2,3	按照windows 操作系统安全基线要求加固vCenter、SSO、VUM所在的	参照OS安全基线检查步骤	参见windows 安全基线和加固方案
访问控制	阻止未使用端口	在W vCe 列明				
其他	配置NTP	为v Ma				
访问控制	禁用datastore浏览	配置 ena				
访问控制	禁用mob	配置 ena				
账户密码	使用service account	使用 务				
证书	限制certificate文件访问	确认 Ser				

vCenter (20项)

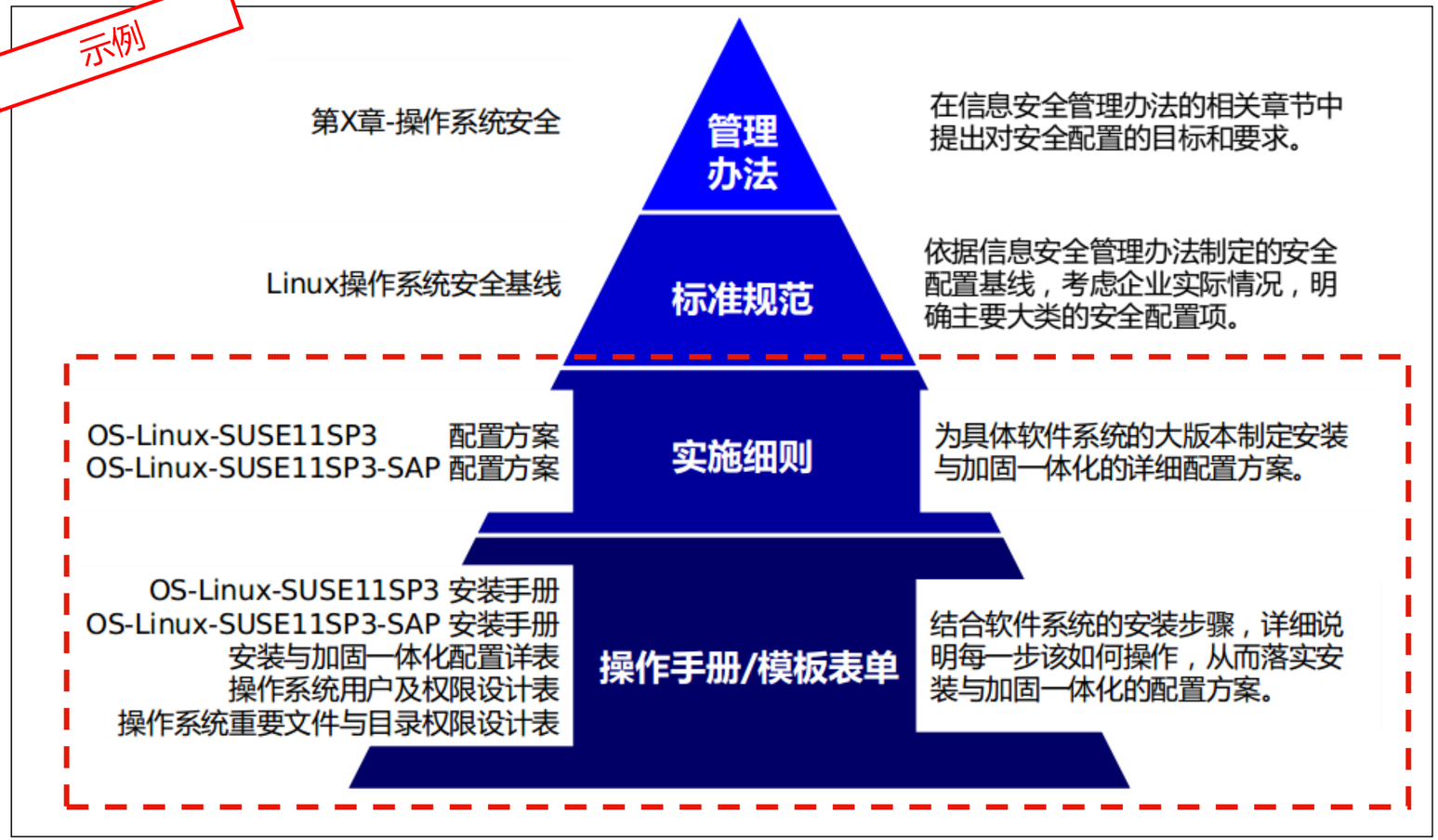
类别	名称	加固要求	风险等级	处置措施	安全判定依据	加固操作参考
其他	禁用Tools自动安装	配置虚拟机vmx配置文件项： isolation.tools.autoInstall.disable值为true	1,2	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-AdvancedSetting -Name "isolation.tools.autoInstall.disable"| Select Entity, Name, Value	通过PowerCli增加配置项，并设置相关参数满足安全基线要求： Get-VM | New-AdvancedSetting -Name "isolation.tools.autoInstall.disable" -value
通信	禁用控制台复制	配置虚拟机vmx配置文件项： 没有isolation.tools.copy.disable，或值为true 没有isolation.tools.dnd.disable，或值为true 没有isolation.tools.SetGUIOptions.enable，或值为false 没有isolation.tools.paste.disable，或值为true	1,2	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-AdvancedSetting -Name "isolation.tools.copy.disable"| Select Entity, Name, Value Get-VM | Get-AdvancedSetting -Name "isolation.tools.dnd.disable"| Select Entity, Name, Value Get-VM | Get-AdvancedSetting -Name "isolation.tools.SetGUIOptions.enable"| Select Entity, Name, Value Get-VM | Get-AdvancedSetting -Name "isolation.tools.paste.disable"| Select Entity, Name, Value 确认各VM对应的值符合安全基线的要求	通过PowerCli增加配置项，并设置相应参数满足安全基线要求： Get-VM | New-AdvancedSetting -Name "isolation.tools.copy.disable" -value $true Get-VM | New-AdvancedSetting -Name "isolation.tools.dnd.disable" -value $true Get-VM | New-AdvancedSetting -Name "isolation.tools.SetGUIOptions.enable" -value $false Get-VM | New-AdvancedSetting -Name "isolation.tools.paste.disable" -value $true
磁盘	禁用磁盘空间收缩	配置虚拟机vmx配置文件项： isolation.tools.diskShrink.disable值为true isolation.tools.diskWiper.disable值为true	1,2,3,	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-AdvancedSetting -Name "isolation.tools.diskShrink.disable"| Select Entity, Name, Value Get-VM | Get-AdvancedSetting -Name "isolation.tools.diskWiper.disable"| Select Entity,	通过PowerCli增加配置项，并设置相应参数满足安全基线要求： Get-VM | New-AdvancedSetting -Name "isolation.tools.diskShrink.disable" -value $true Get-VM | New-AdvancedSetting -Name
通信	禁用主机客户机文件传输	配置虚拟机vmx配置文件项： isolation.tools.hgfsServerSet.disable值为true	1	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-AdvancedSetting -Name "isolation.tools.hgfsServerSet.disable"| Select Entity, Name, Value 确认各VM对应的值符合安全基线的要求	通过PowerCli增加配置项，并设置相应参数满足安全基线要求： Get-VM | New-AdvancedSetting -Name "isolation.tools.hgfsServerSet.disable" -value $true
磁盘	禁用独立非持久模式磁盘	在web client中确认各虚拟磁盘为持久模式，或独立非持久模式； 确认虚拟机vmx配置文件中scsi[0-9]:[0-9].mode值为Persistent或Independent-Persistent	1,2,	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置 3）对云管理平台创建磁盘格式提出要求	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-Harddisk | Select Parent, Name, Filename, DiskType, Persistent 确认各VM对应的值符合安全基线的要求	通过PowerCli增加配置项，并设置相应参数满足安全基线要求： Get-VM | Get-HardDisk | Set-HardDisk [Persistent | Independent-Persistent]
		虚拟机vmx配置文件vmci0.unrestricted为false	1,23	1）使用Power更新已经交付虚拟机 2）更新模板，统一配置虚拟机配置	通过PowerCli查看虚拟机相关配置项： Get-VM | Get-AdvancedSetting -Name "vmci0.unrestricted"| Select Entity, Name, Value 确认各VM对应的值符合安全基线的要求	通过PowerCli改变所有虚拟机相关设置： Get-VM | New-AdvancedSetting -Name "vmci0.unrestricted" -value $false

虚拟机 (19项)

纵深防御落地 —— 在安全基线上更进一步

结合私有云架构与设计要求，梳理符合功能要求的最小化组件清单，明确关键配置，设计用户及权限体系，确定内核/基础系统安全配置与组件/服务的安全配置，确保系统安全达到并超越企业安全基线的要求。

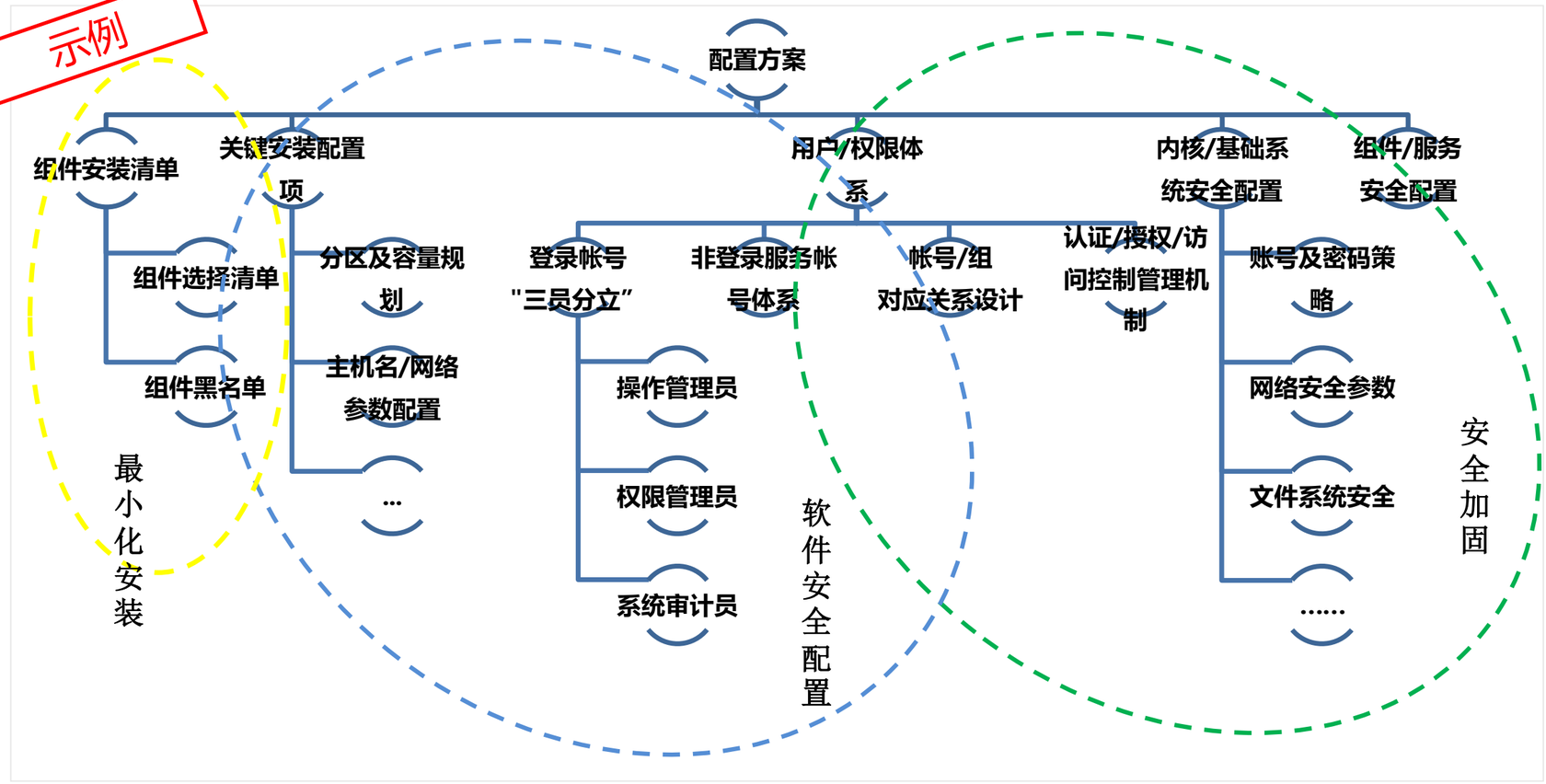
示例

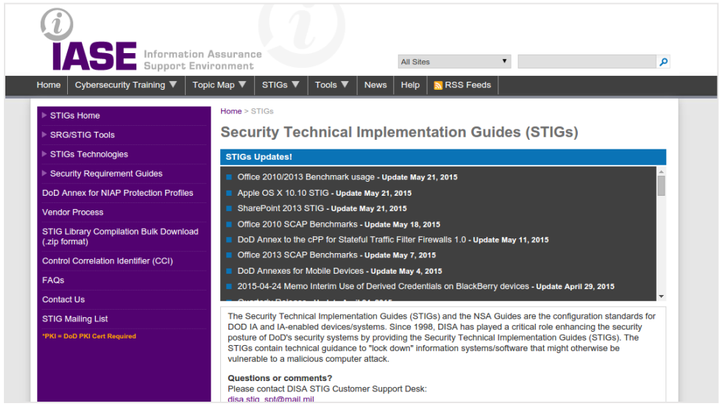


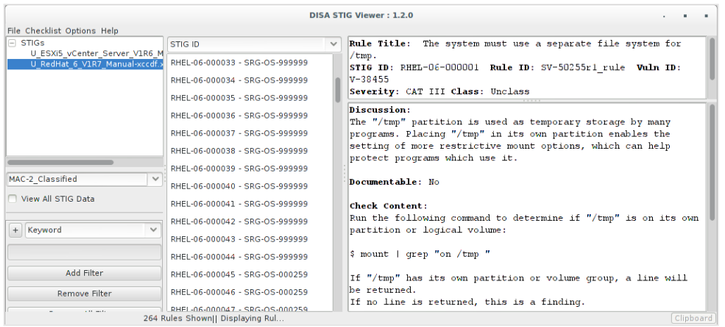
纵深防御落地——“安装、配置与加固”一体化

- 面向软件系统安装配置实施过程，明确各步骤需要结合的安全措施，与安装配置步骤共同形成一体化的方案；
- 实施方案内容与实施目标紧密结合，清晰明确地指导实施工程师开展工作。

示例







STIG提供多安全级别的加固指南，而私有云底层平台行为较固定，且风险较高，可参照STIG中较高安全级别的安全措施进行加固。

纵深防御落地 —— “困住上帝之手”

云管理平台通过在服务器系统上安装代理程序管理目标系统，并通过部署各类SHELL脚本来实现各种自动化运维功能。因此在企业私有云中，需要对代理程序进行严格安全审核并限制权限，同时从脚本的执行权限设计和脚本安全编写规范二个方面来提高云管理平台管理SHELL脚本的安全。

代理程序安全审核

1. 已知安全威胁静态匹配

2. 漏洞威胁分析

3. 安装流程分析

3.1 BMC 降权方案分析

4. 内存分析

4.1 异常进程审计

4.2 系统常规审计

4.3 文件属性异常和 rootkit 检测

4.4 内存取证

基于自由开源软件 Vola 对部署 BMC agent 的机器进行内存 dumpcore 以后的分析：

```
rscd            995    TERM=linux OLDPWD=/ LD_LIBRARY_PATH=/opt/bmc/bladelogic/RSCD/lib PATH=/sbin:/usr/sbin:/bin:/usr/bin RUNLEVEL=3 runlevel=3 PWD=/opt/b
mc/bladelogic/RSCD LANGSH_SOURCED=1 LANG=en_US.UTF-8 PREVLEVEL=N previous=N CONSOLETYPE=vt SHLVL=2 UPSTART_INSTANCE= UPSTART_EVENTS=runlevel UPSTART_JOB=rc R
SCDHOME=/opt/bmc/bladelogic/RSCD _=bin/rscd bin/rscd
rscd            998    TERM=linux OLDPWD=/ LD_LIBRARY_PATH=/opt/bmc/bladelogic/RSCD/lib PATH=/sbin:/usr/sbin:/bin:/usr/bin RUNLEVEL=3 runlevel=3 PWD=/opt/b
mc/bladelogic/RSCD LANGSH_SOURCED=1 LANG=en_US.UTF-8 PREVLEVEL=N previous=N CONSOLETYPE=vt SHLVL=2 UPSTART_INSTANCE= UPSTART_EVENTS=runlevel UPSTART_JOB=rc R
SCDHOME=/opt/bmc/bladelogic/RSCD _=bin/rscd bin/rscd
rscd            999    TERM=linux OLDPWD=/ LD_LIBRARY_PATH=/opt/bmc/bladelogic/RSCD/lib PATH=/sbin:/usr/sbin:/bin:/usr/bin RUNLEVEL=3 runlevel=3 PWD=/opt/b
mc/bladelogic/RSCD LANGSH_SOURCED=1 LANG=en_US.UTF-8 PREVLEVEL=N previous=N CONSOLETYPE=vt SHLVL=2 UPSTART_INSTANCE= UPSTART_EVENTS=runlevel UPSTART_JOB=rc R
SCDHOME=/opt/bmc/bladelogic/RSCD _=bin/rscd bin/rscd
```

审计名称	结果
恶意代码	正常
进程环境	正常
进程关系	正常
文件描述符	正常
内核进程共享权限	正常
系统调用完整性	正常

结论：经过对内存 coredump 的取证分析，未发现明显恶意行为。

脚本执行权限设计

1. 脚本执行权限设计原型

在生产环境中有很多脚本，有一些脚本是需要 root 权限才能完成功能，在这样一种场景下的

脚本安全编码规范

2. 脚本安全编码规范

2.4 尽量避免危险传递参数。

2.5 尽量避免使用 eval

2.6 获得 EUID 的正确方法

```
MYEUID="$(/usr/bin/id -u)"
```

2.7 获得 RUID 的正确方法

```
MYUID="$(/usr/bin/id -u -r)"
```

2.8 获得用户名

```
MYUID="$(/usr/bin/id -u -n)"
```

9，获得 home 目录

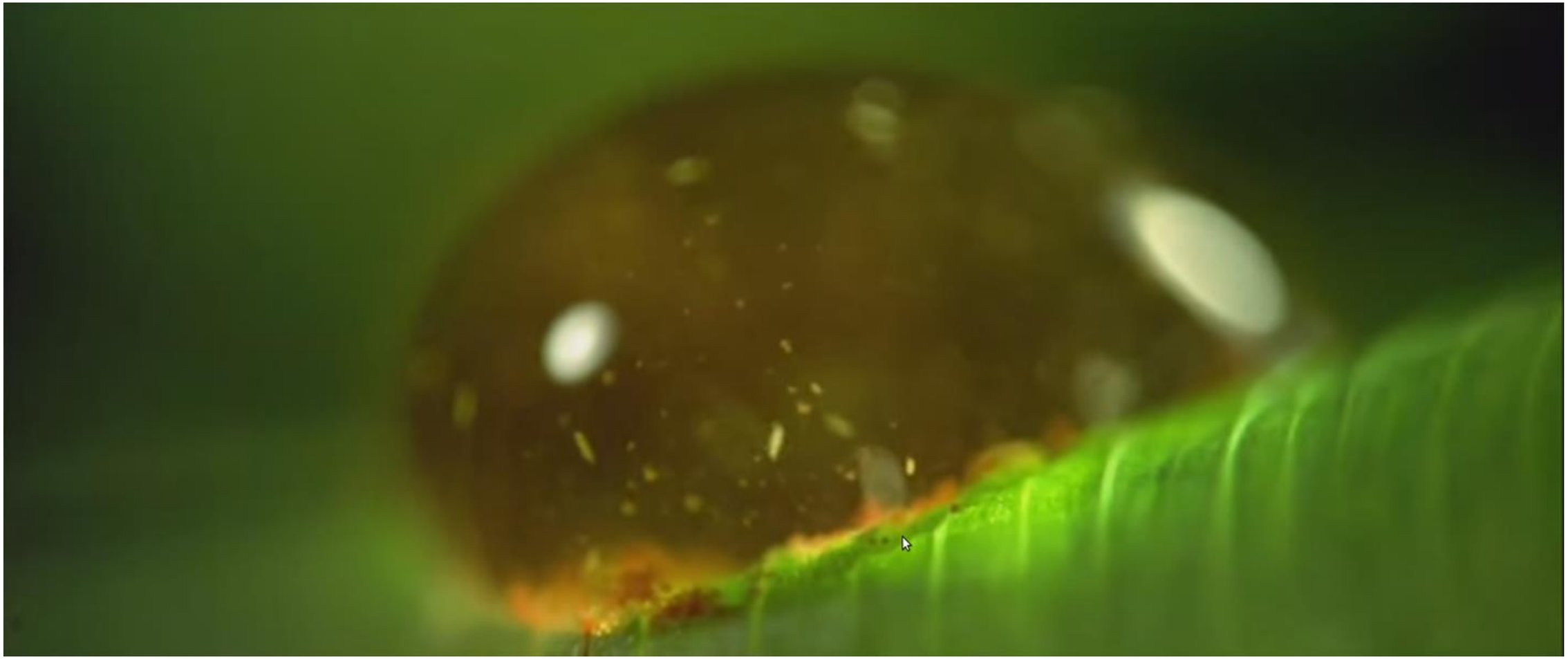
```
HOMEDIR="$(dscl . -read /Users/dg NFSHomeDirecto
```

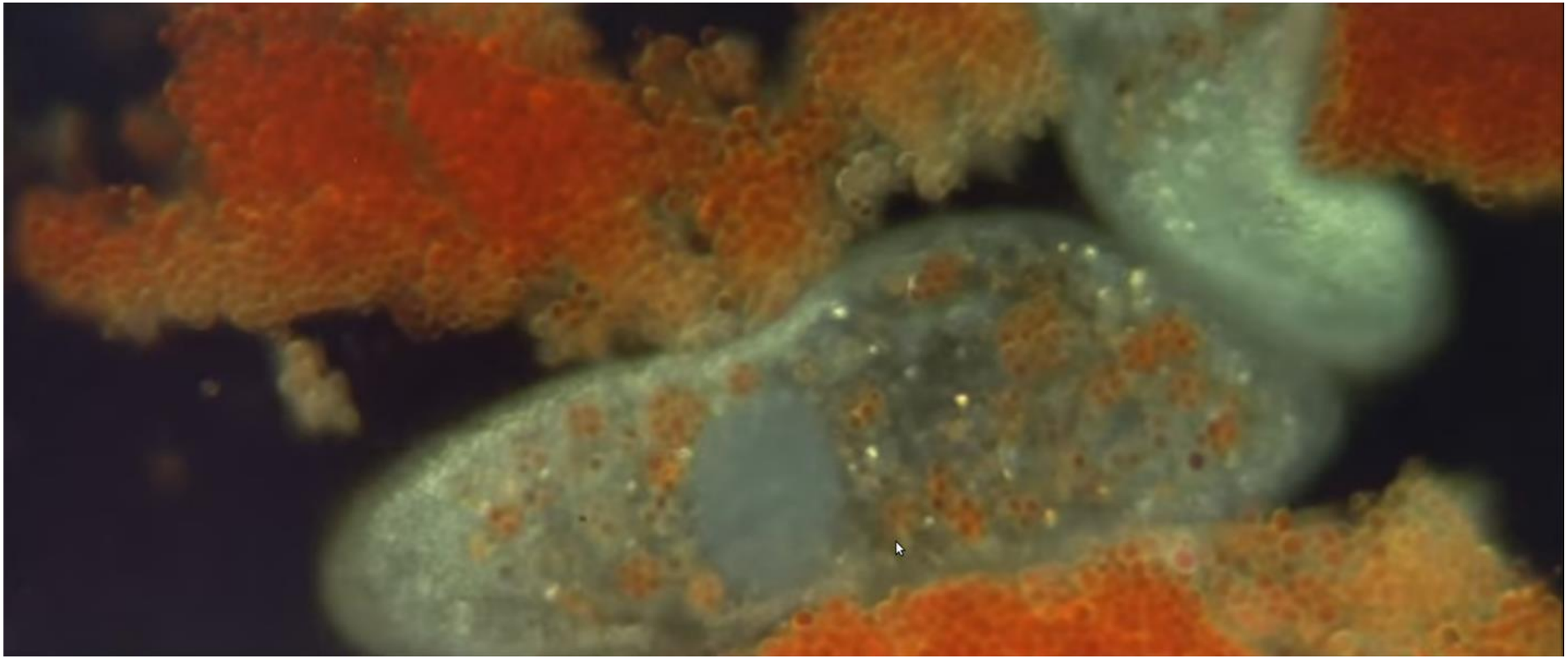
私有云安全防护还需要更多的防御纵深

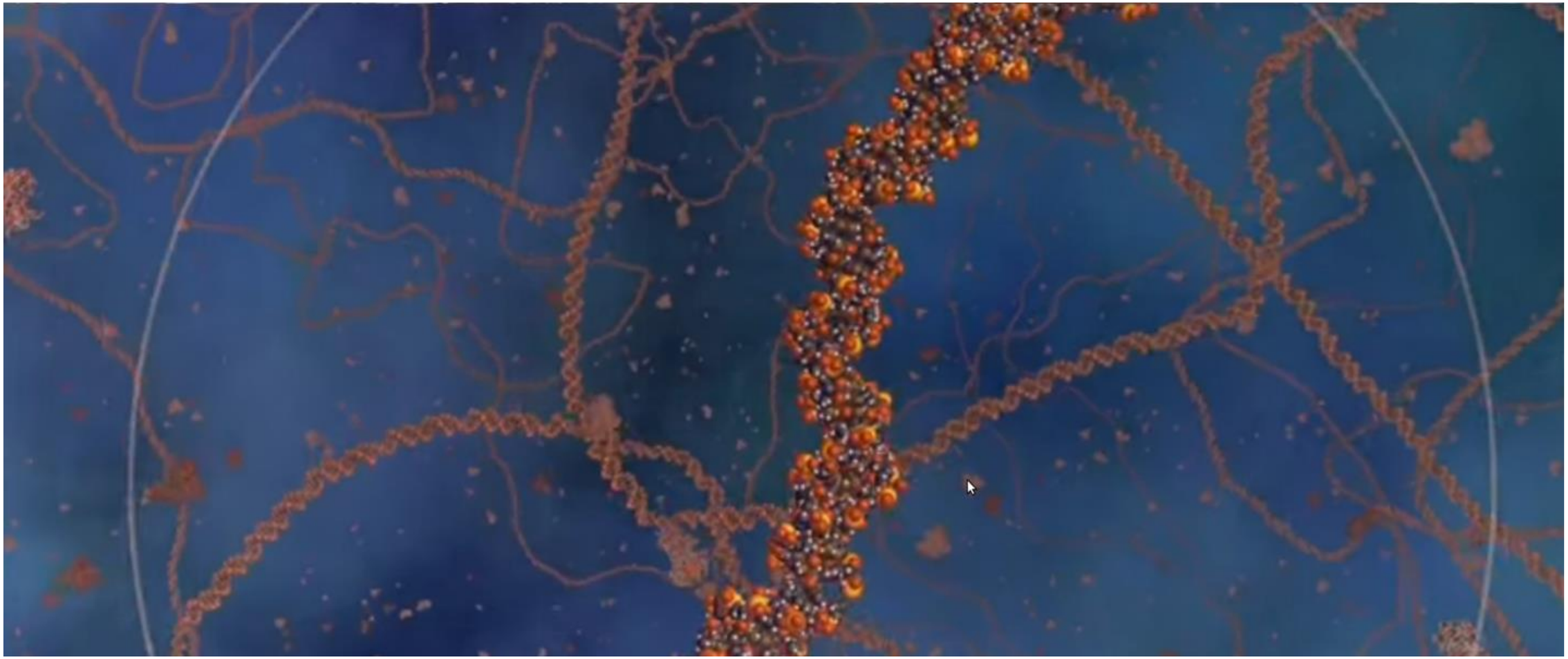
- 构建“安全数据平面”，收集多样化的安全信息数据，结合大数据流式分析技术，对私有云平台进行全面的持续监控；
- 基于SDN技术构建“流网络平台”，提升“东西向”的隔离颗粒度与强度，以及加强云内流量监控；
- 操作系统加固技术在私有云底层平台的应用，特别是通过安全手段固化底层行为；
- 注重私有云技术体系的“生态安全"，防范“技术生态圈”导入的安全威胁；
- 正视底层平台运维可能面对的高级安全威胁，探索将运维文件交换区机制和未知威胁沙箱检测技术结合；
- 面向业务操作与业务数据的云安全代理机制；
- …

Part 03

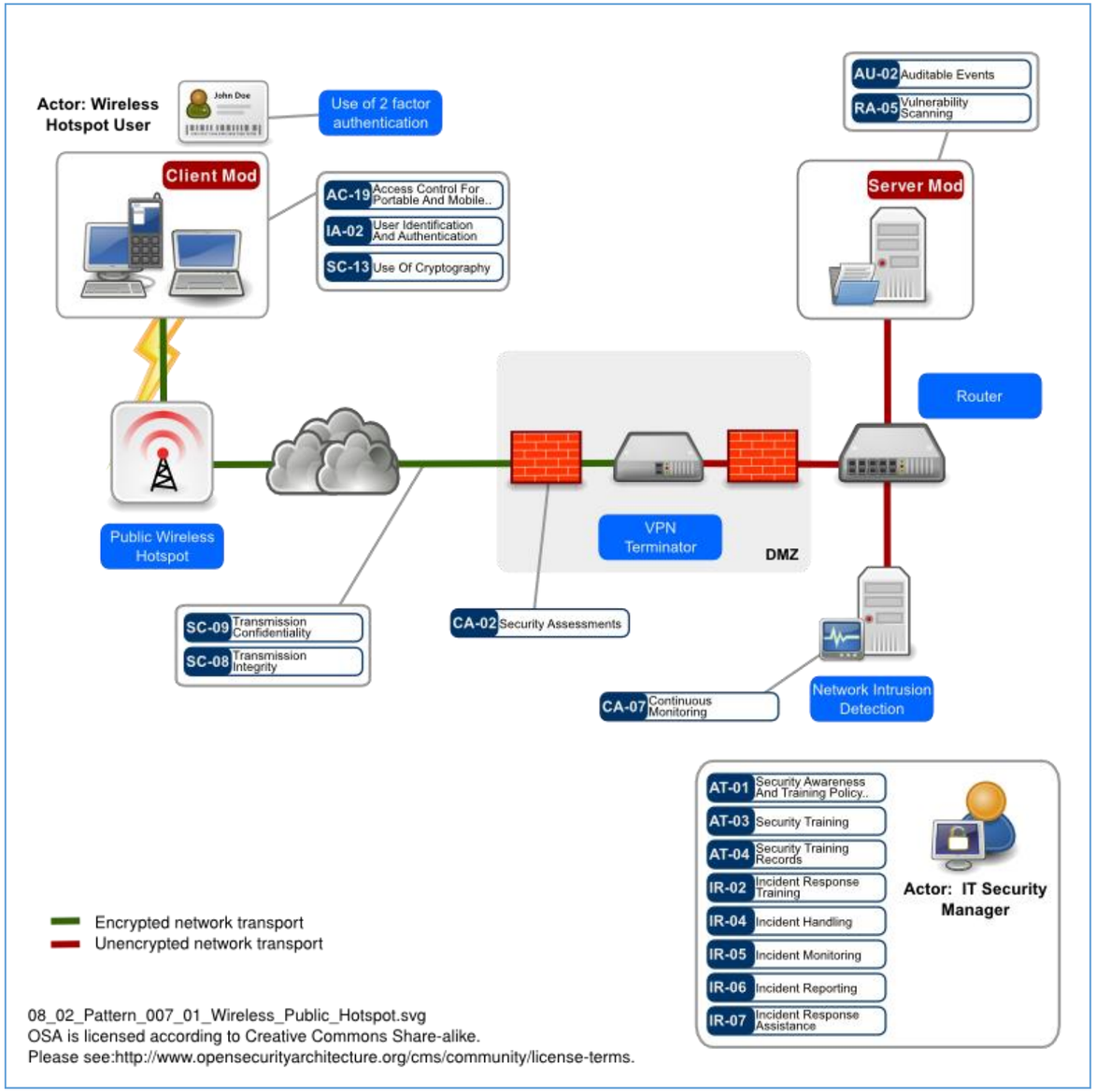
“塔防”纵深防御实践的展望

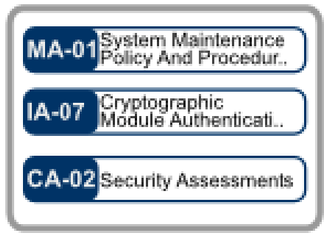






借鉴安全设计模式的形式推动塔防落地





1) Agree baseline

Controls are mapped against governance best practice, security standards, laws and regulations

2) Choose Patterns

Patterns are generated for common security needs encountered in Industry verticals like Finance, Manufacturing and IT.

3) Optimise design

Architecture is optimised for the environment it will operate in.

谢谢大家

THANK YOU FOR YOUR ATTENTION

姓名 电话 邮箱

情报的支撑 塔防的实践