网络空间安全

系统安全

国家网络安全学院

系统安全

1.1 系统安全基本要求

系统安全：确保以电磁信号为主要形式的、在计算机网络化系统进行自动通信、处理和利用的信息内容，在各个物理位置、逻辑区域、存贮和传输介质中，处于动态和静态过程中的机密性、完整性、可用性、可审查性和抗抵赖性的，与人、网络、环境有关的技术安全、结构安全和管理安全的总和。

基本要求：

- 可用性：保证授权用户对系统信息的可访问和使用。
- 完整性：保护信息不被未经授权的实体更改和破坏。
- 机密性：保护信息不受未经授权的访问和泄漏。

安全＝Security + Safety.

- Security是指阻止人为恶意地对安全的危害。
- Safety是指阻止非人为对安全的危害。

系统安全

1.2 系统安全的定义

物理安全：计算机与网络的设备硬件自身的安全，就是信息系统硬件的稳定性运行状态。

运行安全：运行过程中的系统安全，就是信息系统软件的稳定性运行状态。

信息安全（数据安全）：信息自身的安全问题，包括对信息系统中所加工、存储和网络中所传递数据的泄露、仿冒、篡改以及抵赖过程所涉及的安全问题。

系统安全

典型事例1：Meltdown & Spectre

- CPU 底层芯片设计漏洞
- 乱序执行和推测执行会引起CPU缓存的污染，从而攻击者可以发起基于cache的测信道攻击偷取数据。
- 攻击范围广，近20年的Intel、AMD、ARM等处理器都有影响，其中对Intel处理器的影响尤为严重

系统安全

1.3 系统安全措施

系统安全措施＝{法律措施，教育措施，管理措施，技术措施…}

注意：决不能低估法律、教育、管理的作用，许多时候它们的作用大于技术。

1.3.1 技术措施

- 技术措施＝{硬件系统安全、操作系统安全、密码技术、网络通信安全、数据库安全、病毒防治技术，防电磁辐射技术，信息隐藏技术，电子对抗技术，容错技术，・・・}

注意：硬件系统和操作系统安全是基础，密码、网络安全等是关键技术。

系统安全

1.3.2 法律措施

系统安全措施包括各级政府关于信息安全的各种法律、法规。

1.3.3 教育措施

对人的思想品德教育、安全意识教育、安全法规的教育等。

国内外的计算机犯罪事件都是人的思想品德出问题造成的。

信息安全是一个系统工程必须综合采取各种措施才能奏效。

1.3.4 管理措施

系统安全管理措施既包括信息设备、机房的安全管理，又包括对人的安全管理，其中对人的管理是最主要的。

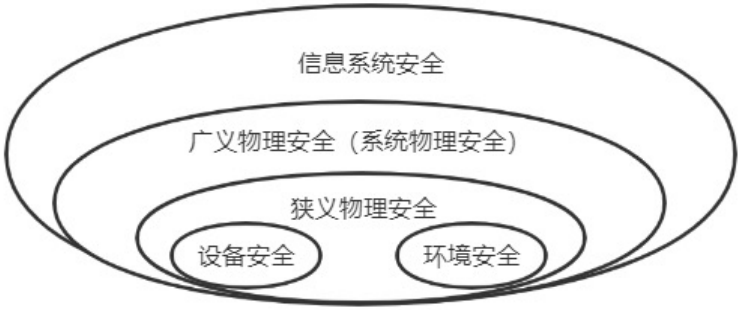
“三分技术，七分管理。”

目前，计算机网络系统安全的最大威胁之一是计算机网络的安全监管。

2、物理安全

2.1 物理安全定义

- 物理安全又叫实体安全（Physical Security），是保护计算机设备、设施（网络及通信线路）免遭地震、水灾、火灾、有害气体和其他环境事故（如电磁污染等）破坏的措施和过程。



物理安全概念体系示意图

- 物理安全主要包括：环境安全、设备安全和介质安全。

2、物理安全

2.2 环境安全、设备安全和介质安全

- 环境安全：系统所在环境的安全，主要是场地与机房。
- 设备安全：主要指设备的防盗、防毁、防电磁辐射泄露、防止线路截获、抗电磁干扰及电源保护等。
- 介质安全：包括介质数据的安全及介质本身的安全。

2、物理安全

2.2.1 环境安全技术

- 安全保卫技术是环境安全技术的重要一环，主要的安全技术措施包括：防盗报警、实时监控、安全门禁等。

- 计算机机房的温度、湿度等环境条件保持技术可以通过加装通风设备、排烟设备、专业空调设备来实现。

- 计算机机房的用电安全技术主要包括不同用途电源分离技术、电源和设备有效接地技术、电源过载保护技术和防雷击技术等。

- 计算机机房安全管理技术是指制定严格的计算机机房工作管理制度，并要求所有入机房的人员严格遵守管理制度，将制度落到实处。

2、物理安全

2.2.2 设备安全的保护内容

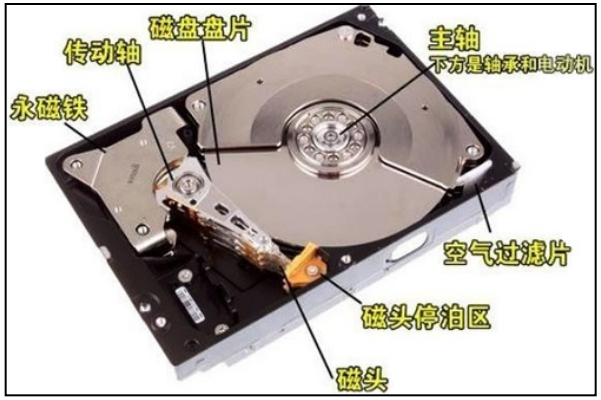
- 设备防盗：使用防盗手段保护计算机系统设备和部件。
- 设备防毁：一是对抗自然力的破坏；二是对抗人为的破坏。
- 防止电磁信息泄露：以提高系统内敏感信息的安全性。
- 防止线路截获：主要防止对计算机信息系统通信线路的截获与干扰。
- 抗电磁干扰：以保护系统内部的信息。
- 电源保护：为计算机信息系统设备的可靠运行提供能源保障

2.2.3 介质安全

- 介质数据的安全：防止记录的信息不被非法窃取、改、破坏或使用。
- 介质本身的安全：防盗、防毁、防霉等

2、物理安全

实际问题：硬盘忽然掉电会损坏硬盘和数据吗？



- 一般操作情况下，盘片在主轴的带动下，以每分钟数千转的速度飞速旋转。传动轴在程序指令下，带着磁头来到指定位置开始读取/写入数据。
- 正常断电的时候，硬盘控制芯片会收到SATA控制器传来的STANDBY IMMEDIATE命令。这时控制器会把磁头归位到磁头停泊区（Parking Zone），停在该区域避免了和磁盘接触。
- 在意外断电的时候，磁盘控制器会利用空气动力和一些电容的余电，将磁头移到着陆区里面降落，从而保证不会划伤盘片。着陆区也是硬盘没事干时，磁头的休息区。

着陆区（Landing Zone），这里并没有磁道，表面介质也不同。

2、物理安全

例1：闪存颗粒（SSD）

- JEDEC组织对SSD定下了标准，如右上图：
- 即消费品在掉电情况下，在30度室温中需要保证1年数据不丢失。它的来源是一份Intel的研究报告，如右下图：
- 在30度情况下，数据经过52周即有可能出现数据丢失。如果我们把存放温度（即断电时保存温度）提高到55度，2周数据就有可能丢失。
- 当然这是最低标准，而且要求很破旧的SSD都要遵守的原则。实际情况会好的多。
- 但是结论仍是不变的：SSD放着不动，数据可能会丢失的！

JEDEC：固态技术协会固态及半导体工业界的一个标准化组织

JEDEC Global Standards for the Microelectronics Industry

SSD endurance classes and requirements

Application Class	Workload	Active Use (power on)	Retention Use power off)	Functional Failure Rqmt (FFR)	UBER
Client	Client	40ºC 8 hrs/day	30ºC 1 year	≤3%	$\leq 10^{-15}$
Enterprise	Enterprise	5[illegible]ºC 24hrs/day	40ºC 3 months	≤3%	$\leq 10^{-16}$

JEDEC Global Standards for the Microelectronics Industry

Temperatures and data retention

- Tables show # weeks retention as a function of active and power-off temperatures.
- Numbers are based on Intel's published acceleration model for the detrapping retention mechanism (the official JEDEC model in JESD47 and JEP122 for this mechanism).

Client

Power Off Temperature \ Active temp	25	30	35	40	45	50	55
25	25	58	65	79	105	155	244	404
30	14	28	32	39	52	76	120	199
35	7	14	16	20	26	38	61	101
40	4	7	8	10	14	20	31	52
45	2	4	4	5	7	10	17	27
50	1	2	2	3	4	6	9	15
55	1	1	1	2	2	3	5	8

Enterprise

Power Off Temperature \ Active temp	25	30	35	40	45	50	55
25	7	9	12	20	33	58	101
30	3	4	6	10	16	28	50
35	2	2	3	5	8	14	25
40	1	1	2	3	4	7	13
45	0	1	1	1	2	4	7
50	0	0	0	1	1	2	4
55	0	0	0	0	1	1	2

Material submitted by Intel

2、物理安全

例2：通信线路安全技术

防止记录的信息不被非法窃取、改、破坏或使用。用一种简单（但很昂贵）的高技术加压电缆，

可以获得通信线路上的物理安全。通信电缆密封在塑料套管中，并在线缆的两端充气加压。线上连接了带有报警器的监示器，用来测量压力。如果压力下降，则意味电缆可能被破坏了，技术人员还可以进一步检测出破坏点的位置，以便及时进行修复。

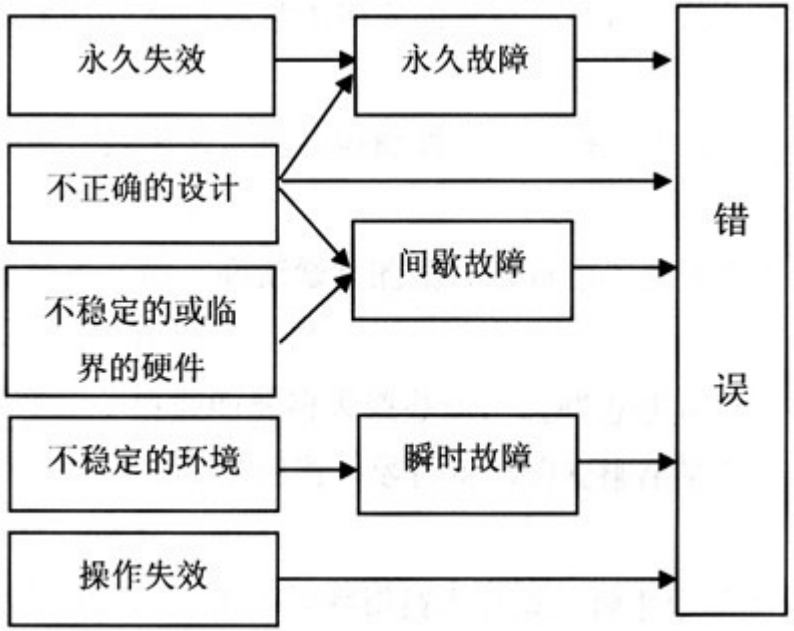
距离大于最大长度限制的系统之间，不采用光纤线通信；或加强复制器的安全，如用加压电缆、警报系统和加强警卫等措施。

2、物理安全

2.3 容错与可靠性

- 失效（failure）是指硬件物理特性异变，或软件不能完成规定功能的能力。
- 故障（fault）是指硬件或软件的错误状态，是失效在逻辑上的等效。一个故障可以用种类、值、影响范围和发生时间来描述。
- 错误（error）是指程序或数据结构中的故障表现形式，是故障和失效所造成的后果。

容错设计的软件可以有某些规定数目的故障但不导致失效但对无容错的软件而言，故障即失效。



2、物理安全

2.3 容错与可靠性

2.3.1 可靠性

- 可靠性的含义
  - 广义：一切旨在避免、减少、处理、度量软件/硬件故障（错误、缺陷、失效）的分析、设计、测试等方法、技术和实践活动。
  - 狭义：指软件/硬件无效运行的定量度量。
- 可靠度：在规定的运行环境中和规定的时间内软件无失效运行的机会。

2.3.2 容错

- 容错（Fault-tolerance）：容忍故障，考虑故障一旦发生时能够自动检测出来并使系统能够自动恢复正常运行。
- 当出现某些指定的硬件故障或软件错误时，系统仍能执行规定的一组程序，或者说程序不会因系统中的故障而中止或被修改，并且执行结果也不包含系统中故障所引起的差错。

2、物理安全

2.3.3 容错技术的发展与现状

- 早在20世纪60年代，美国国家航空航天局(NASA)开始支持容错计算机的研制。1961年-1965年研制出第一台星载计算机,采用三余度表决，关键部位采用四余度。1962年-1969年研制出用于阿波罗登月飞船的容错计算机,性能指标为连续工作250h，可靠性达99%。
- 在20世纪90年代，北京控制工程研究所就开发了卫星控制系统地面实时故障诊断专家系统SCRDES,有效提高了地面故障诊断的自动化程度。
- 当前中国的航天器型号已经不同程度地实现了在轨故障诊断与重构。例如，"环境减灾-1A、1B"卫星能够利用星上的硬件冗余和解析冗余实现部件的故障诊断，并且可以进行系统级的安全重构。

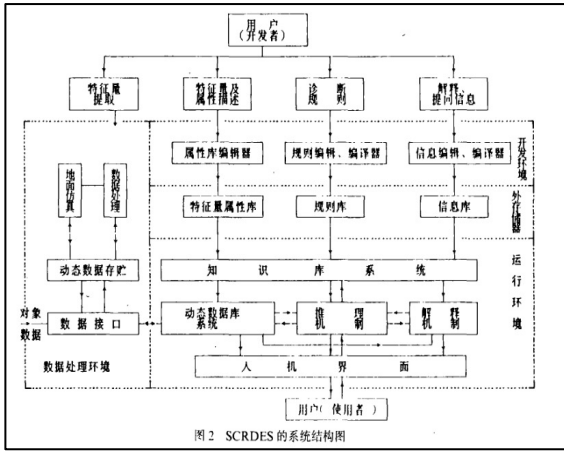


图2 SCRDES的系统结构图

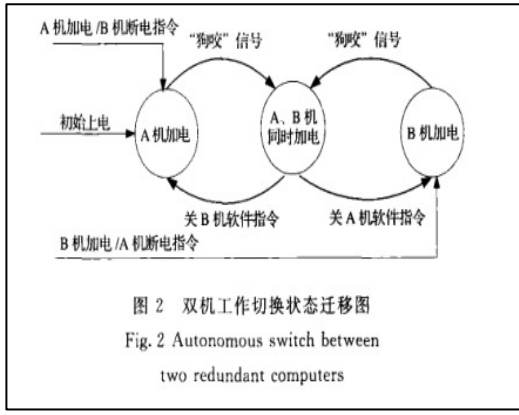


图2 双机工作切换状态迁移图

Fig. 2 Autonomous switch between two redundant computers

2、物理安全

2.3.4 容错与实现方法

- 容错技术是指在一定程度上容忍故障的技术，也称为故障掩盖技术（Fault Masking）。采用容错技术的系统称为容错系统。
- 容错主要依靠冗余设计来实现，它以增加资源的办法换取可靠性。
- 冗余技术可分为：硬件冗余、软件冗余、信息冗余和时间冗余。

2、物理安全

2.3.4 容错与实现方法

冗余技术：以增加资源的办法换取可靠性。

- 硬件冗余：在常规设计的硬件之外附加备份硬件，包括静态冗余、动态冗余。
- 软件冗余：用于测试、检错的外加程序。
- 信息冗余：增加信息的多余度，使其具有检错和纠错能力。
- 时间冗余：重复地执行指令或一段程序而附加额外的时间。

2、物理安全

2.3.4 容错与实现方法

硬件容错包括：

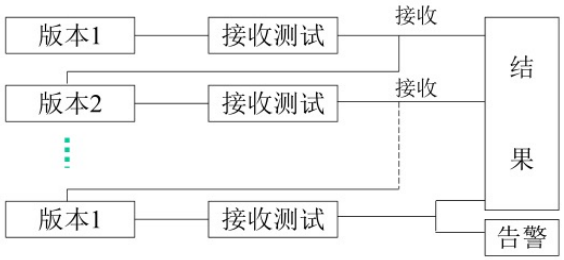
- 硬件备份（硬件堆积冗余，待命储备冗余，混合冗余系统等）；
- 数据备份（完全备份，差分备份，增量备份，按需备份）（本地备份，异地备份）；
- 双机容错系统（一个CPU出现故障，其他CPU保持继续运行）；
- 双机热备份（“心跳线”保持主系统与备用系统的联系）；
- 三机表决系统（表决器根据3台主机的运行结果进行表决）；
- 集群系统（均衡负载的双机或多机系统）。

2、物理安全

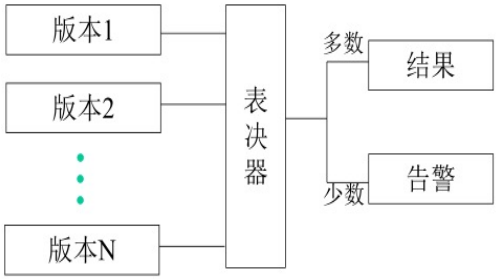
2.3.4 容错与实现方法

软件容错包括：

- 恢复块方法：具有相同功能的一个主块和几个后备块，一个块就是一个执行完整的程序块，主块首先投入执行，结束后进行验收测试，如果没有通过，系统经现场恢复后由一后备块运行。
- N版本程序设计：由N个实现相同功能的不同程序同时（或几乎同时）在松耦合计算机上运行，然后比较运行结果，在出现不一致的情况下，利用多数表决决定一个最优先的结果
- 防卫式程序设计：在程序中包含错误检查代码和错误恢复代码，使得一旦错误发生，程序能撤销错误状态，恢复到一个已知的正确状态中。



恢复块技术



N版本程序设计

2、物理安全

2.3.4 容错与实现方法

N版本程序设计的应用：

- 航天飞机的飞行控制软件；
- 飞机（如空中客车A310、A330）的机翼控制系统；
- 铁路编组和站点交通信号（如瑞典等国）的控制系统；
- 原子核反应堆控制系统等。

2、物理安全

2.3.5 冗余技术实例

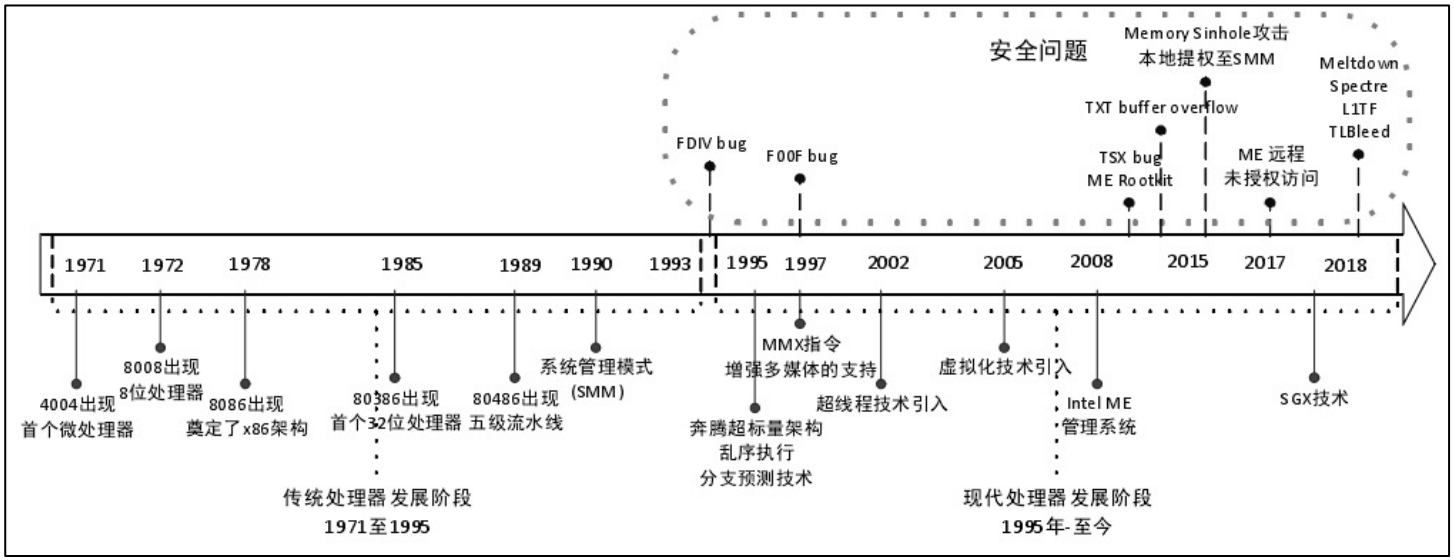
- 基于虚拟化的双机热备份（简称双机热备）是一种常见的利用冗余来提高系统可用性的技术。双机热备的基本思想是：备份节点对工作节点的运行状态进行实时备份，当工作节点发生故障时，备份节点立刻接管工作节点上运行的服务，保证系统服务的连续性。

- RemuS是由英国哥伦比亚大学研发的一种虚拟机双机热备技术，通过在备份节点上建立工作节点副本并实时更新的方式为运行在Xen虚拟化平台上的虚拟机提供热备服务。RemuS方案默认的数据复制间隔为20ms，每次复制采用增量式数据复制方法，只对复制间隔期间修改过的内存页面和磁盘块进行复制，可在保证数据精确备份的同时缩短复制数据占用的时间。

3、硬件与芯片安全

3.1 芯片安全的概念

- 随着SoC芯片规模以及复杂度的提高，在设计时大量采用第三方IP集成，且OEM代工、分离制造和测试封装等生产方式，使得在整个芯片的制造周期都存在安全隐患，如被植入硬件木马或存在未知系统漏洞。
- 特别是当芯片中存在运行时触发的恶意电路或在运行时遭到软硬件联合攻击，将对芯片功能以及数据安全造成严重破坏。

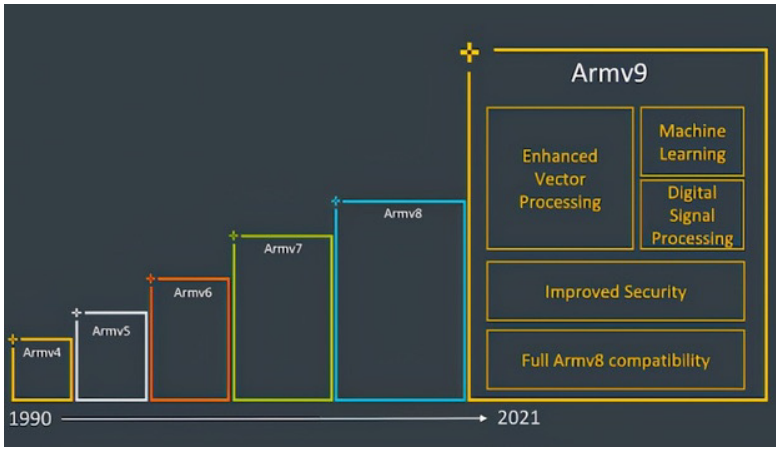


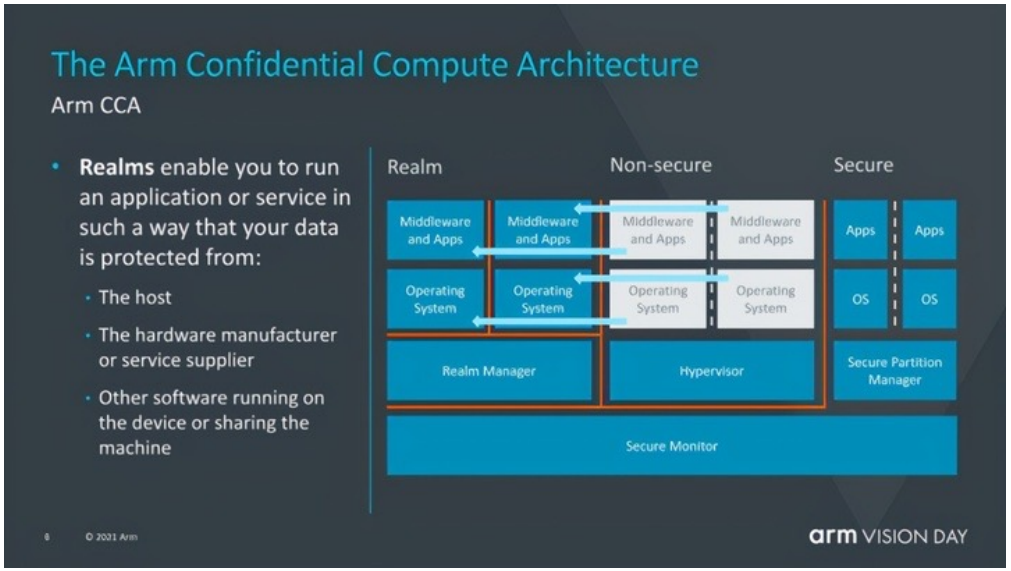
3、硬件与芯片

3.2 芯片安全研究现状

3.2.1 ARM架构

➢2021年，ARM公司正式宣布推出全新的Armv9架构，除了常规的性能和指令集架构升级，关于安全性方面，Armv9引入Realms机密计算模块，将敏感应用和操作系统隔离在Realm中；Realm比机密虚拟机更加通用，既支持机密虚拟机形态，也支持机密操作系统形态。



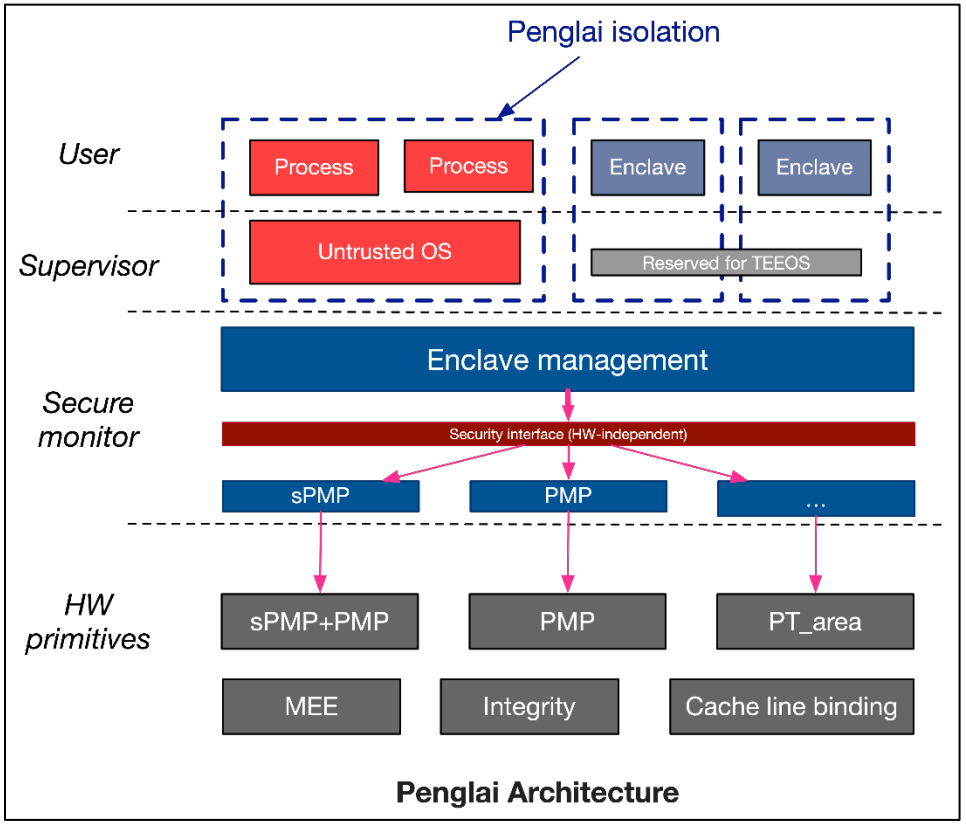


3、硬件与芯片安全

3.2 芯片安全研究现状

3.2.2 RISC-V架构

- 2021年，上海交通大学并行与分布式系统研究所与瓶钵信息科技开源了基于RISC-V 架构的全新TEE安全系统“蓬莱” 。
- 蓬莱扩展了现有RISC-V硬件原语，通过软硬件协同的方式来支持隔离环境的可扩展性。
- 具体地，蓬莱系统实现一套新的RISC-V指令扩展sPMP（特权级物理内存保护机制），允许在TEE OS中或者Secure monitor中实现可扩展的物理内存隔离。



3、硬件与芯片安全

3.2 芯片安全研究现状

3.2.3 x86架构

- SGX：Intel SGX是Intel在原有架构上增加了一组新的指令集和内存访问机制，它允许应用程序实现一个被称为enclave的容器，在应用程序的地址空间中划分出一块被保护的区域，为容器内的代码和数据提供机密性和完整性的保护，免受拥有特殊权限的恶意软件的破坏。

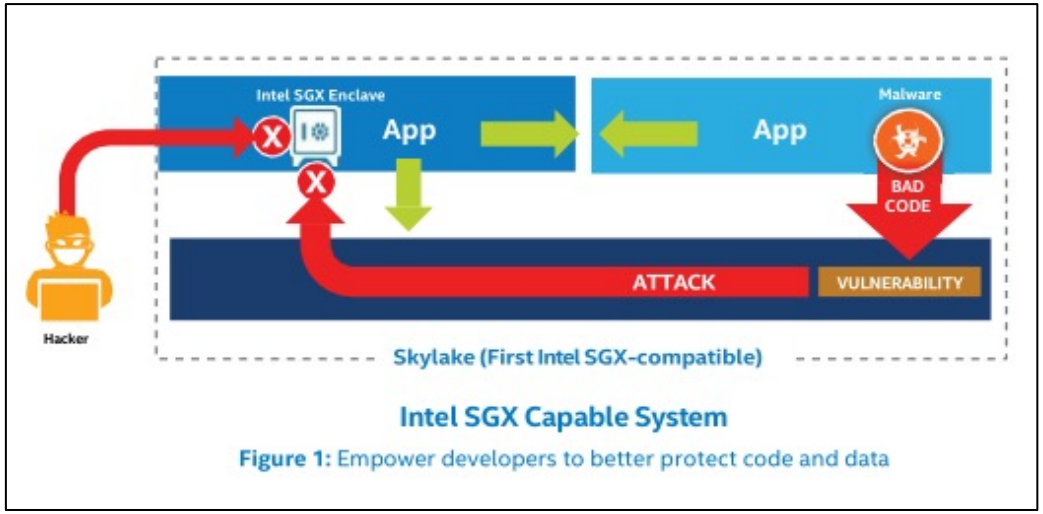


Figure 1: Empower developers to better protect code and data

3、硬件与芯片安全

3.3 芯片安全实例

3.3.1 硬件木马

- 硬件木马是指在IC设计或制造过程中被蓄意植入或更改的特殊电路模块、或者是设计者无意留下的设计缺陷。
- 当其以某种方式被激活后，可能改变IC的功能或规格，泄漏敏感信息，造成IC的性能下降、失去控制，甚至是不可逆的破坏。

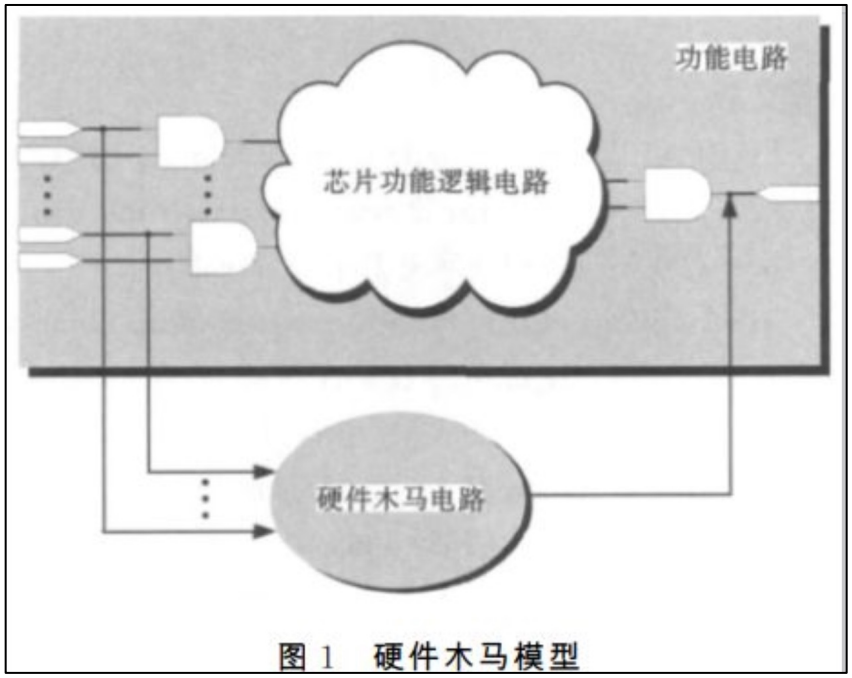


图 1　硬件木马模型

3、硬件与芯片安全

3.3 芯片安全实例

3.3.1 硬件木马

- 在2016年的IEEE安全与隐私大会上，来自密歇根大学的研究人员详细描述了一种微型硬件后门的概念验证攻击—A2。
- 其基本原理是利用电容的特性，当某个被某条指令触发的某个逻辑信号触发譬如几千次之后，即可使电容充电到阈值，使后面的逻辑输出翻转，执行一个譬如说提权的操作。

2016 IEEE Symposium on Security and Privacy

A2: Analog Malicious Hardware

Kaiyuan Yang, Matthew Hicks, Qing Dong, Todd Austin, Dennis Sylvester
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI, USA
{kaiyuan, mdhicks, qingdong, austin, dmcs}@umich.edu

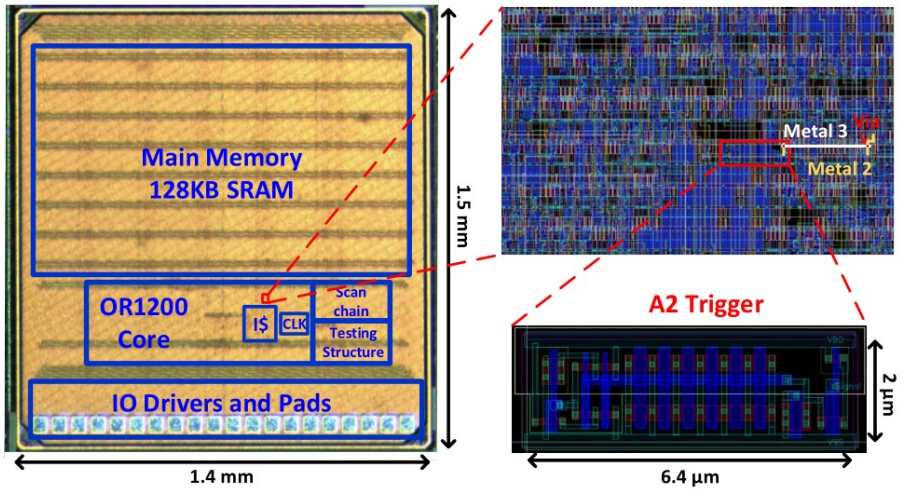


Figure 15: Die micrograph of analog malicious hardware test chip with a zoom-in layout of inserted A2 trigger.

3、硬件与芯片安全

3.3 芯片安全实例

3.3.2 CPU漏洞

- 1994年，美国弗吉尼亚州的教授Thomas Nicely在使用计算机处理长除法时意外发现了奔腾处理器中的浮点除错误(FDIV bug)，该错误的原因是处理器内置的乘法表中存在输入错误。
- 1997年，奔腾F00F错误(F00F bug)在奔腾P5微架构系列处理器上被发现，该漏洞可导致处理器死锁停止运行，其内在原因是指令编码无效同时异常处理机制被死锁。
- 2009年，在BlackHat会议上，研究人员发现了英特尔可信执行技术(Intel TXT,Intel Trusted Excution Technology)的漏洞，通过此漏洞可以绕过Intel TXT的可信保护的启动过程。
- 2015年，Domas Christopher发现了名为“Memory Sinkhole"的漏洞并在BlackHat会议上公开，通过高级可编程中断控制器地址可重映射的特性，结合系统管理模式的固有代码可以使权限由Ring0层提升至Ring2层。

3、硬件与芯片安全

3.2 芯片安全实例

3.2.2 CPU漏洞

- 2018年1月，Google Project Zero团队对外公布了熔断(Meltdown)漏洞,该漏洞影响了所有采用了乱序执行技术的intel处理器，除了Itanium系列及2013以前的Atom外几乎1995年以来所有Intel处理器均受影响，部分ARM处理器也不例外。
- 利用此漏洞，普通用户可以读取计算机中所有的内存信息，会导致口令及敏感信息的泄露，同时对虚拟化技术下的云平台各租户的敏感信息也造成威胁。

3、硬件与芯片安全

3.3 芯片安全实例

3.3.2 CPU漏洞

- Meltdown漏洞出现的同时，Google Project Zero团队公布了幽灵(Spectre)漏洞，此漏洞几乎影响了所有的现代处理器，涉及的厂商包括Intel. AMD以及ARM。
- 利用此漏洞，普通用户可以读取计算机中所有的内存信息，会导致口令以及敏感信息的泄露，同时威胁到虚拟化下云平台各租户的敏感信息的安全性。
- Spectre漏洞和Meltdown漏洞同样是基于侧信道的信息泄露型漏洞，区别在于Spectre漏洞在于现代处理器中采用的用于提升处理器效能的分支预测机制存在缺陷所致。

4、可信计算

如何提高微机的安全性？

➢ 我们的理解：

硬件系统安全和操作系统安全是信息系统安全的基础，密码技术、网络安全技术等是关键技术。

根据以上学术观点：只有从芯片、主板等硬件结构和BIOS、操作系统等底层软件作起，综合采取措施，才能比较有效的提高微机系统的安全性。

➢ 我们的观点：

可信≈可靠+安全

用词：Dependable and Trusted Computing

可信计算机系统是能够提供可信计算服务的计算机软硬件实体，它能够提供系统的可靠性、可用性、主体行为与信息的安全性。

4、可信计算

4.1 可信计算的基本概念

可信计算思想：

- 首先建立一个信任根。

  信任根的可信性由物理安全和管理安全确保。

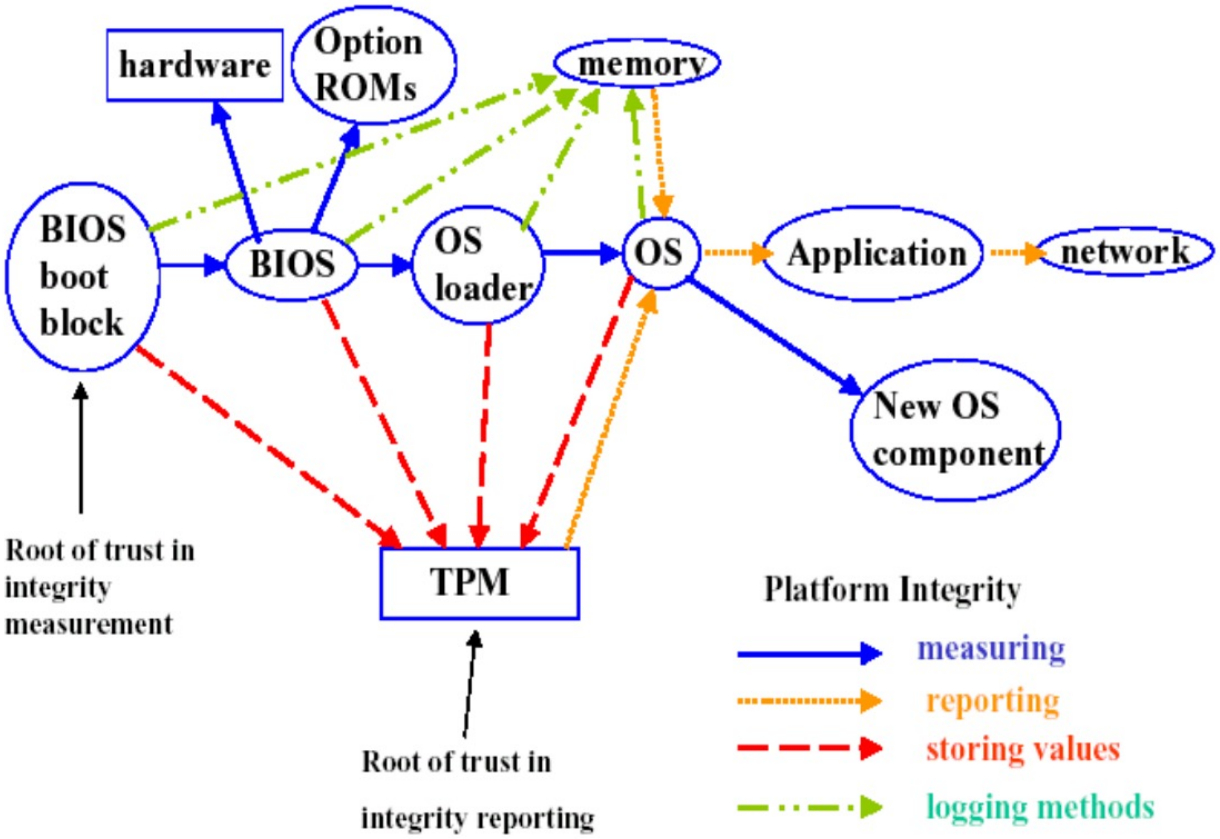
- 再建立一条信任链。

  从信任根开始到硬件平台、到操作系统、再到应用，一级认证一级，一级信任一级。从而把这种信任扩展到整个计算机系统。

- 可信计算的思想源于社会。

4、可信计算

4.3 可信计算的实现方法



4、可信计算

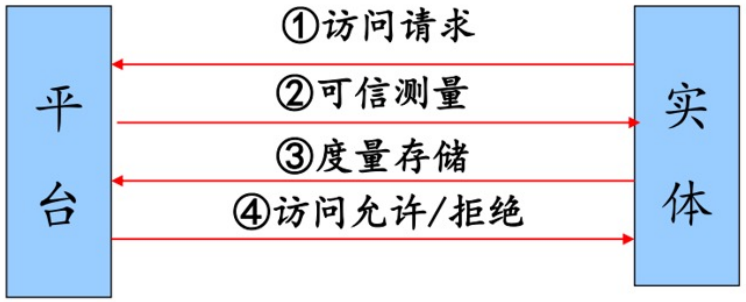
4.3 可信计算的实现方法

➢可信测量技术

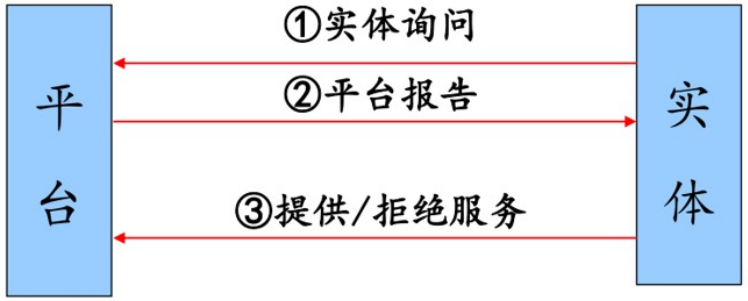
- 可信的测量：任何想要获得平台控制权的实体，在获得控制权之前都要被测量，判断其是否可信。
- 度量的存储：对实体可信的测量以及该过程的审计信息将被TPM保存，以此向访问实体报告平台或其上运行实体的可信度的依据。
- 度量的报告：需要知道平台可信状态的实体，在获得许可后，可以得到当前TPM中保存的测量值的报告。询问实体据此来衡量当前平台的可信度，并决定是否与该平台建立会话。

4、可信计算

4.3 可信计算的实现方法



实体访问平台，平台测量实体



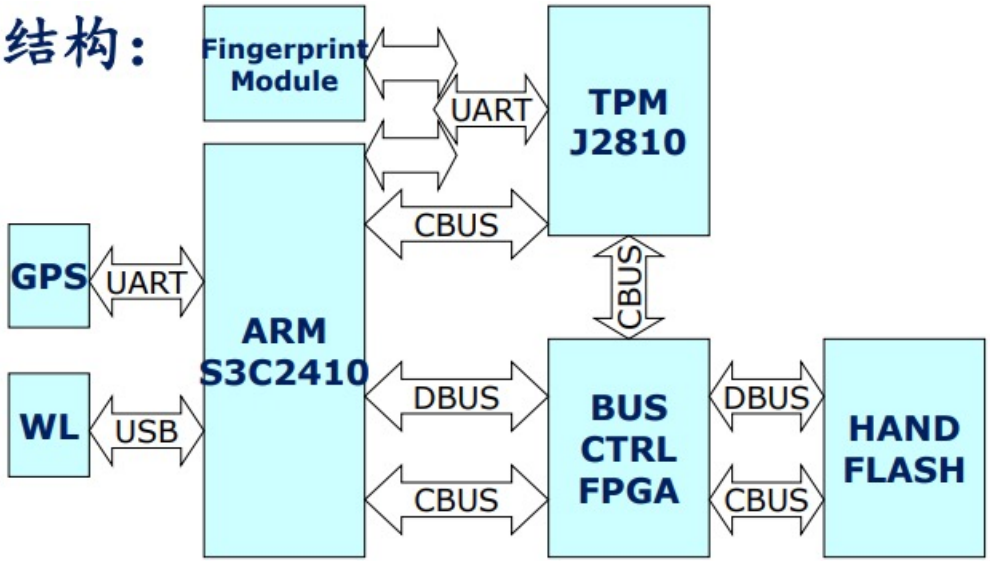
实体询问平台，平台提供报告

4、可信计算

4.4 可信计算平台

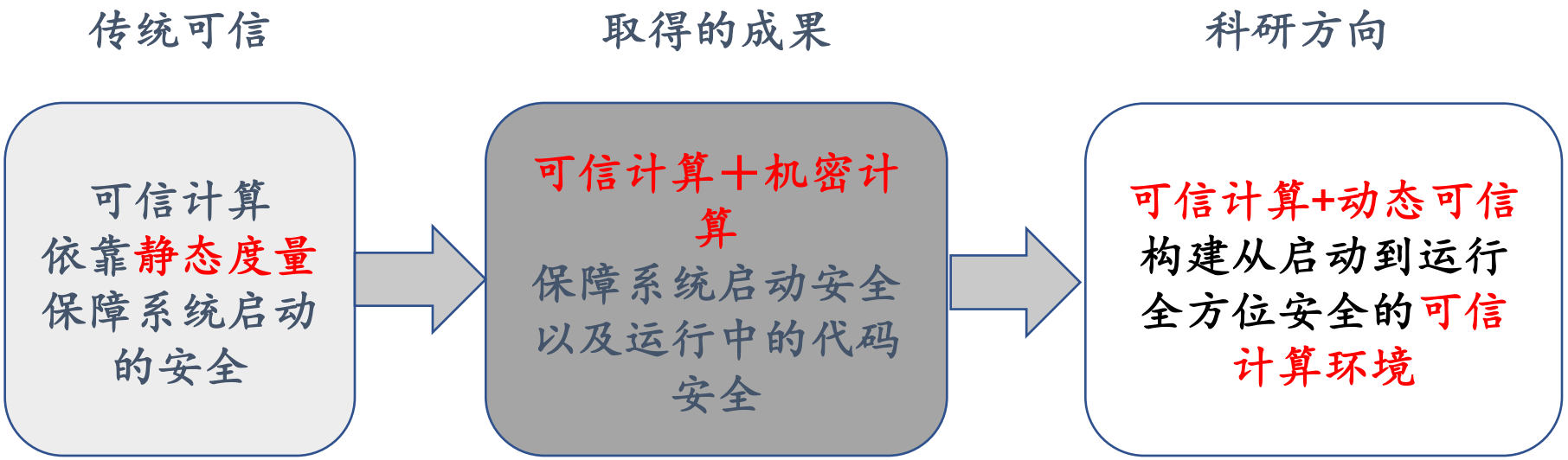
- 可信PDA

系统结构：



4、可信计算

4.5 可信计算环境



5、操作系统安全

5.1 操作系统安全的概念

➢ 安全操作系统(Secure Operating System)是指对所管理的数据与资源提供适当的保护级，有效地控制硬件与软件功能的操作系统。安全操作系统在开发完成后，在正式投入使用之前,一般都要求通过相应的安全性评测。

➢ 操作系统安全(Operating System Security) 是指操作系统无错误配置、无漏洞、无后门、无特洛伊木马等，能防止非法用户对计算机资源的非法存取，一般用来表达对操作系统的安全需求。

5、操作系统安全

5.2 操作系统面临的安全威胁

安全威胁是指这样一种可能性,即对于一定的输入，经过系统处理，产生了危害系统安全的输出。威胁大多是通过利用操作系统和应用服务程序的弱点或缺陷实现的。

按照形成安全威胁的途径，将操作系统的安全威胁分为：

- 不合理的授权机制
- 不恰当的代码执行
- 不恰当的主体控制
- 服务的不当配置
- 网络协议的安全漏洞
- 不安全的进程间通信(IPC)

5、操作系统安全

5.3 安全操作系统发展

5.3.1 国外安全操作系统发展

- 60年代:美等发达国家开始从事安全操作系统的研究。
- 70年代:关国等发达国家开发出一些安全系统应用于军方和政府部门。
- 80年代- 90年代,关国等发达国家依据TCSEC标准成功开发了B1,B2,B3,A1级的原型系统或实用系统。
- 2001年代-至今,英国和欧盟发达国家依据CC标准成功开发了达到EAL4,EAL5等以上评估保证级的实用系统或产品。

5、操作系统安全

5.3 安全操作系统发展

5.3.2 国内安全操作系统发展

➢1990年以来，我国参照TCSEC B1，B2级和GB17859第三级、第四级进行安全操作系统的研究和探讨：

- 1996年，海军计算技术研究所实现国产自主的安全增强包。
- 1998年，电子工业部十五所根据TCSEC标准的B1级安全要求，对Unix操作系统的内核进行了安全性增强。
- 江苏南大苏富特软件股份有限公司开发完成了基于Linux的安全操作系统SoftOS。
- 中国科学院软件所基于红旗Linux操作系统，实现了符合我国GB17859—1999第三级要求的安全功能。
- 此外，国防科技大学、总参56所等其他单位也开展了安全操作系统的研究与开发工作。

5、操作系统安全

5.4 安全操作系统实例

Multics.

1965年，由美国贝尔实验室和麻省理工学院的MAC项目组一起联合开发。目标是实现并发访问信息存储系统时的高安全性，包括:受控共享、用户鉴别、用户间隔离、超级用户保护、对用户写权限程序和特定特权的控制等。

结果不太理想，但迈出了安全操作系统研究重要的第一步，比如:其中Bell和Lapadula合作设计的BLP机密性安全模型，就是首次提出并成功地应用于Multics系统。

5、操作系统安全

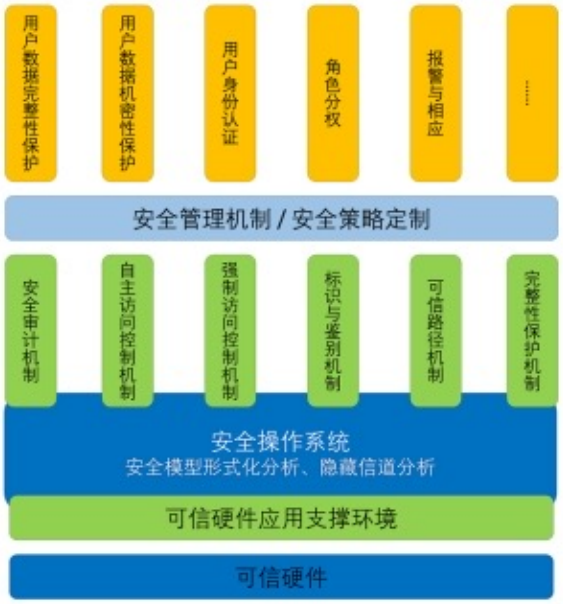
5.4 安全操作系统实例

方德方舟安全操作系统

满足GB/T 20272-2006《信息安全技术操作系统安全技术要求》第四级结构化保护级的要求，保证服务器在可控、安全、高效状态下运行，能够为政府、军工、金融、证券、涉密等领域的用户提供自主可控的基础计算平台。

方德方舟安全操作系统安全机制：

- 基于三权分立的管理机制
- 强化的身份标识与认证机制
- 强化的身份标识与认证机制
- 基于内核层的安全审计
- 具有良好的软硬件兼容性



5、操作系统安全

5.5 主流操作系统安全的解决方案

操作系统安全性的主要目标是标识系统中的用户，对用户身份进行认证，对用户的操作进行控制，防止恶意用户对计算机资源进行窃取，篡改，破坏等非法存取，防止正当用户操作不当而危害系统安全，从而既保证系统运行的安全性，又保证系统自身的安全性。具体包括如下几个方面：

- 身份认证机制
- 访问控制机制
- 数据保密性
- 数据完整性
- 系统的可用性等；
- 审计

操作系统安全的核心是访问控制，主要有自主访问控制、强制访问控制、基于角色的访问控制等。

5、操作系统安全

5.5 主流操作系统安全的解决方案

5.5.1 Windows操作系统安全方案

Windows操作系统是微软研发的一套操作系统，已经拥有了30多年的悠久历史，目前熟知的Windows操作系统有Windows 95、Windows 98、Windows 2003、Windows XP、Windows Vista、Windows 7、Windows 10等，该系统在历史的发展中也在不断持续更新和维护。

Windows10的主要安全机制：

- 安全登陆
- 权限保护
- bootlocker加密
- 组策略
- 防火墙
- UEFI安全启动机制(Secure Boot)
- Windows Defender机制

5、操作系统安全

5.5 主流操作系统安全的解决方案

5.5.2 Linux操作系统安全方案

Linux是开源的免费操作系统，因其开源的特性，系统的漏洞更容易被发现，也更容易被修补，Linux主要被应用于各种服务器，通常通过命令行进行人机交互，也有桌面操作系统，比如redhat、ubuntu等等发行版。

Linux的主要安全机制：

- PAM(Pluggable Authentication Modules)认证机制
- 文件系统加密
- 入侵检测机制
- 安全日志文件机制
- 访问控制
- 防火墙机制

系统管理员可以通过PAM认证机制选择合适的认证方法，Linux通过SELinux(Secure Enhanced Linux)实现强制访问控制，允许管理员更加灵活地定义安全策略。

5、操作系统安全

5.5 主流操作系统安全的解决方案

5.5.3 Android操作系统安全方案

Android是一种基于Linux的、自由的、开源的操作系统。它主要使用于移动设备，如智能手机和平板电脑，由Google公司和开放手机联盟开发。

Android系统主要的安全机制：

- Android进程沙箱隔离机制
- 应用程序签名机制
- 权限声明机制
- 访问控制机制
- 高效进程间通信机制Binder
- 内存管理机制

Android的内存管理机制可以确保当内存不足时，自动清理最低级别进程所占用的内存空间，且具备清理不再使用共享内存区域的能力，应用程序的APK文件必须被开发者数字签名，防止恶意软件替换安装的应用。

5、操作系统安全

5.6 鸿蒙操作系统

鸿蒙系统（Harmony OS）是由华为公司历时9年基于微内核开发的场景分布式操作系统，与Android系统及iOS系统是用完全不同框架的手机操作系统。

鸿蒙系统是全球首个基于微内核全场景分布式开发的操作系统，其目标是将分布式能力最大化造就新硬件、新交互、新服务，建立超级虚拟终端互联世界，并且基于Android生态架构开发的应用软件可直接迁移至鸿蒙系统，仅需下载鸿蒙系统安装包就可平稳、安全的完成迁移及部署。鸿蒙系统在实现智能服务的同时，也充分体现微内核多元性和稳定性。

基于微内核的鸿蒙os架构

5、操作系统安全

5.7 银河麒麟操作系统

我国国内信创操作系统领域麒麟系、统信UOS 领跑，麒麟系操作系统在信创政府领域内具备较大优势。麒麟家族包括中标麒麟（NeoKylin）、银河麒麟（Kylin）、优麒麟（Ubuntu Kylin）和麒麟信安（Kylinsec）。其中中标麒麟和银河麒麟占主要地位。

2020年8月，中国电子发布银河麒麟操作系统V10 。其拥有性能领先、生态丰富、体验提升、云端赋能、融入移动、内生安全六大优势。

银河麒麟桌面操作系统

6、软件安全

6.1 软件安全问题

随着计算机网络的迅速发展和软件的广泛应用，软件的安全性已经成为备受关注的一个方面，成为关系到金融、电力、交通、医疗、政府等各个领域的关键问题。

在当前黑客肆虐，病毒猖獗的网络环境下，越来越多的软件因为自身存在的安全漏洞，造成黑客以及病毒攻击的对象，给用户带来严重的安全隐患。

软件的安全问题是从软件诞生之日起就存在的，但当其成为大众性商品被广泛应用时，其安全问题才得到足够的重视。

当前的软件安全问题可粗略地分为：计算机病毒、恶意软件、软件漏洞、软件后门。

6、软件安全

6.1 软件安全问题实例

实例1：2021年01月27日，据360CERT监测发现RedHat发布了sudo 缓冲区/栈溢出漏洞的风险通告，该漏洞编号为CVE-2021-3156。

漏洞的具体危害：

在解析命令行参数的方式中发现了基于堆的缓冲区溢出。任何本地用户（普通用户和系统用户）都可以利用此漏洞，而无需进行身份验证，攻击者不需要知道用户的密码。成功利用此漏洞可以获得权限。

6、软件安全

6.1 软件安全问题实例

实例2：OpenSSL被曝存在拒绝服务高危漏洞，漏洞编号为CVE-2021-3449。

关于OpenSSL高危漏洞的预警通报

发布时间：2021-04-21 20:41

近日，OpenSSL被曝存在拒绝服务高危漏洞，漏洞编号为CVE-2021-3449。OpenSSL是一个开放源代码的软件库包，使用加密算法、证书等提供安全通信功能，目前广泛应用于网页（web）服务器。在OpenSSL TLS服务器启用TLSv1.2和重新协商功能情况下，攻击者可从客户端发送恶意构造的ClientHello请求触发该漏洞，从而导致服务器拒绝服务。目前官方已确认并修复该漏洞，受影响的版本是OpenSSL 1.1.1-1.1.1j。

鉴于该漏洞影响范围大，潜在危害程度高，建议各单位立即组织排查，及时将OpenSSL升级至安全版本OpenSSL 1.1.1k（https://openssl.org/），堵塞漏洞，消除隐患！

6、软件安全

6.1 软件安全问题的解决方案

➢软件漏洞分析

漏洞从发现到产生实际危害的整个过程可分为漏洞挖掘、漏洞分析、漏洞利用三个阶段。

漏洞挖掘分为如下阶段：

- 传统漏洞挖掘技术
  - ✓静态漏洞挖掘技术
  - ✓动态漏洞挖掘技术
- 基于学习的智能化漏洞挖掘技术

6、软件安全

6.1 软件安全问题的解决方案

➢软件漏洞分析

静态漏洞挖掘是指在不运行目标程序的前提下分析目标程序(源代码或二进制)的词法、语法和语义等,并结合程序的数据流、控制流信息,通过类型推导、安全规则检查、模型检测等技术挖掘程序中的漏洞。

主要工具有面向C/C+源码的Cppcheck，FlawFinder、面向PHP源码的RIPS、面向JAVA源码的FindBugs等。

优点：直接对目标程序进行分析不需要构造程序的执行环境，能提取较为完整的控制流等信息,可能发现动态漏洞挖掘技术难以发现的漏洞。

缺点：一方面，由于静态漏洞挖掘技术往往依赖于人工构造的漏洞模式,对先验知识依赖性较大，另一方面,因为无法获得程序实际动态运行过程中的上下文信息,静态漏洞挖掘技术具有精度低、误报率高的缺陷。

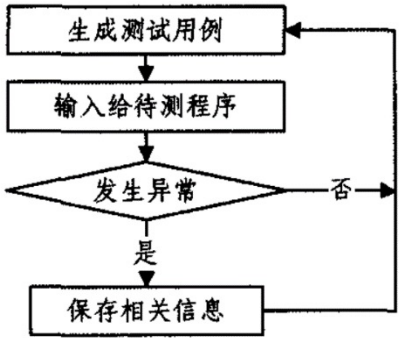
6、软件安全

6.1 软件安全问题的解决方案

➢软件漏洞分析

动态漏洞挖掘技术通过动态监测程序在给定环境中运行时的行为，可以准确地发现异常的发生，从而找出目标程序的漏洞。**模糊测试和动态污点分析**是两种典型的动态漏洞分析技术。

动态分析方法有精确度高，定位准的优点。但是，也存在一些问题，如模糊测试存在依赖于输入种子的质量、测试攻击面模糊、难以发现访问控制漏洞和设计逻辑错误等问题。



自动化模糊测试流程

6、软件安全

6.1 软件安全问题的解决方案

➢软件漏洞分析

随着深度学习的发展，**基于学习的智能化漏洞挖掘技术**成为了人们关注的热点，其过程可以概括为：收集大量软件程序相关数据，分析软件代码中的控制流、抽象语法树、数据流等信息，送入机器学习模型进行训练，最终得到一个用于进行漏洞分类或预测的分类器模型，用于漏洞检测。

优点：基于学习的智能化漏洞挖掘技术具有更高的准确率和完备性，能满足实际生产中对大型复杂软件系统进行漏洞挖掘的需求，且消耗的人力资源相对较少。

缺点：基于学习的智能化漏洞挖掘技术的代码粒度通常较粗，无法确定漏洞的确切位置，且该方法的效果受限于算法本身，有不可解释性等问题。

6、软件安全

6.1 软件安全问题的解决方案

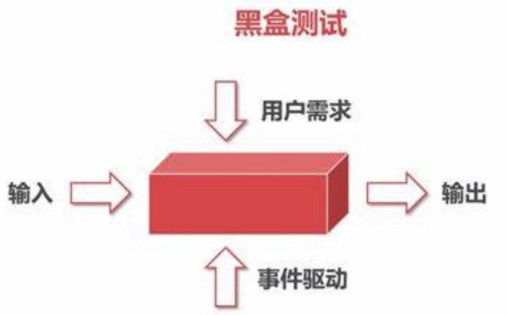
➢病毒对抗技术

病毒对抗主要研究病毒的防护病毒的清除等。病毒的检测技术主要有特征值检测技术、校验和检测技术、虚拟机技术、主动防御技术等。同时，人们可以通过现象观察法和使用反病毒软件检算机是否感染病毒，也可采用感染实验法分析新的病毒。

常规计算机病毒的检测是将主引导区、可能染毒的文件和内存空间与病毒特征库中的病毒标记进行对比分析，从而检测特定的病毒。

➢黑盒测试

黑盒测试时完全不考虑程序内部的结构和处理过程，只按照规格说明书的规定来检查程序是否符合它的功能要求。黑盒测试是在程序接口进行的测试，又称为功能测试。

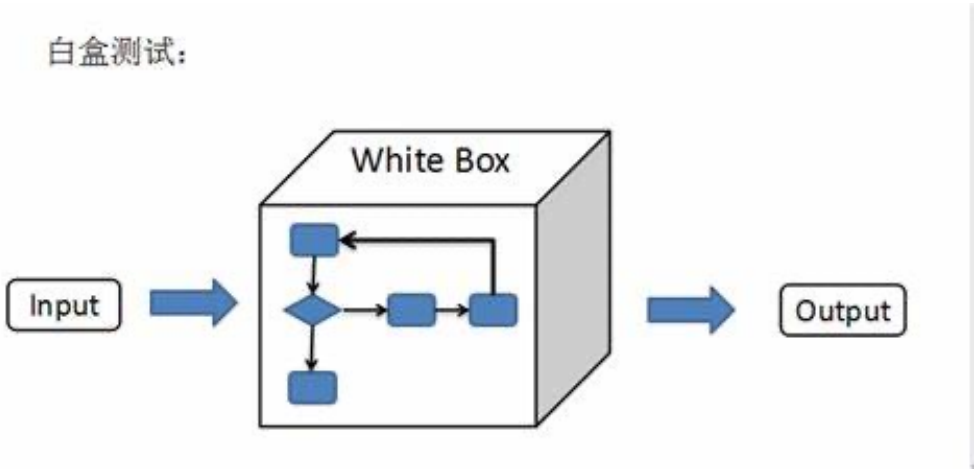


6、软件安全

6.1 软件安全问题的解决方案

➢ 白盒测试

白盒测试时将程序看作是一个透明的盒子，也就是说测试人员完全了解程序的内部结构和处理过程。所以测试时按照程序内部的逻辑测试程序、检验程序中的每条通路是否都能按预定的要求正确工作。白盒测试又称为结构测试。



7、数据库安全

7.2 数据库安全

7.2.1 数据库安全问题

数据库因包含有各种有价值的敏感信息，例如金融或知识产权信息、公司数据、个人用户数据等等，一直是黑客攻击的目标，黑客企图通过破坏服务器、数据库来获利。数据库安全的重要性体现在以下方面：

- 数据库是重要的应用软件。
- 数据库集中存储和管理着大量的重要数据，如军事、 政治、金融等数据。数据库成为不法分子攻击的主要目标。
- 数据库要支持查询、插入、删除、更新等操作，而且 存储的数据量大、时间长是其重要特点。
- 数据库的安全措施应适应数据库的特点。。

7、数据库安全

7.2 数据库安全

7.2.1 数据库安全问题

数据库常见的安全问题如下：

- 物理数据库的完整性：

保证数据库系统中的数据不受各种自然或者物理问题而破坏，如地震、水灾、火灾、盗窃、电力问题或设备故障等。

- 逻辑数据库的完整性：

对数据库的结构化特征提供保证，确保数据库系统结构、数据库模式数据库数据不被非法修改，事物处理及操作符合数据库各种完整性约束。

- 元素安全性：

确保数据库各种存储元素满足机密性、完整性、可用性等限制。元素控制比文件控制复杂，拥有更多的粒度层次和更灵活的安全策略。

7、数据库安全

7.2 数据库安全

7.2.1 数据库安全问题

- 可审计性：

  可以提供追踪存取和修改数据库元素的用户的能力。记录数据库中所有事物和操作，保留详细的审计和日志记录，提供有效地威慑和事后追查、分析和取证工具。审计和日志的粒度直接决定审计的时间和代价。

- 访问控制：

  确保只有授权用户和程序可以访问那些允许它们访问的数据元素，同时保证对不同的用户限制使用不同的控制策略并允许灵活设置。

- 身份认证：

  不允许一个未经授权的用户对数据库进行操作。

- 可用性：

  数据库系统能够随时对授权用户提供高质量的数据库服务，让用户能够最大限度地访问允许他访问的数据。

7、数据库安全

7.2 数据库安全

7.2.1 数据库安全问题

- 推理控制：

数据推理是指用户通过合谋、拼凑等方式，从合法获得的低安全等级信息及数据中推导出受高安全等级保护的内容，也可以进一步估计数据推理的准确程度。推理控制机制必须保证用户不能从被公开发布的、授权可被访问的信息以及统计信息中，推导出秘密的、未被授权访问的信息以及统计信息，保护所有的秘密信息。

- 多级保护：

多级保护是信息系统等级安全中重要的思想。根据现实应用的要求，可以将数据划分为不同密级的集合，也可以将同一记录中的不同字段划分为不同的保密等级，还可以将同一字段的不同值划分为不同的安全等级，从而实现数据的等级划分以及用户依据相应等级安全策略要求的等级访问。

7、数据库安全

7.2 数据库安全

7.2.1 数据库安全问题

- 推理消除隐通道：

在多级安全模型中，隐通道是一种违反系统安全策略，表面上合法的操作序列，它是一种可被攻击者利用于将高等级数据向低等级用户传送的通信信道。消除隐通道的目的是防止程序或者用户之间通过非法授权进行信息传递，需要发现各种隐通道包括时间隐通道、存储隐通道等。

7、数据库安全

7.2 数据库安全

7.2.2 数据库安全实例

➢数据泄露：

2021年，WizCase安全团队在扫描FBS外汇交易平台时发现了严重的数据泄露事件，数以百万计的机密记录，包括用户姓名、账号密码、电子邮件地址、护照号码、信用卡、交易数据等信息可能落入不法分子手中。

FBS是成立于2009年的国际外汇交易公司，在全球190个国家/地区拥有超过40万合作伙伴和1600万名交易员，是世界上最受欢迎的在线外汇交易平台之一。

FBS有一个不安全的ElasticSearch对外暴露，其中包含近20TB 的数据（超过160亿条记录）。该服务器没有任何密码保护，其中的财务数据可以自由访问，这是极其危险的。近20TB的数据遭到泄漏，涵盖160亿条记录，全球的数以百万计的FBS用户受到影响。泄漏的信息包括了用户基础信息、平台账户信息，甚至财务数据。

7、数据库安全

7.2 数据库安全

7.2.2 数据库安全实例

➢数据库漏洞：

2019年，某实验室在数据库漏洞挖掘方面又有重要新发现。这一次发现了新类型的国际数据库漏洞——Oracle Database Server高危漏洞，目前已经得到Oracle确认，并分配CVE编号（CVE-2019-2444）。

CVE-2019-2444是一个使用符号链接攻击手段，达到本地提权效果的漏洞，攻击者可由oracle用户权限提升至root用户权限该漏洞属于提权漏洞，一旦利用了漏洞，能够使得非权限用户获得权限提升，从而通过一台机器看到数据库里的所有数据，包含实例。

7、数据库安全

7.2 数据库安全

7.2.3 数据库安全解决方案

- 数据库审计：

数据库是个黑盒子，对来自内网、外网的用户和系统对核心数据的访问情况，尤其是违规访问情况缺乏可视化，

数据库审计分析数据库的网络活动，对访问数据的访问进行展示。

- 数据库防火墙：

设置对核心数据的访问规则，这种访问规则是独立于且不同于数据库自身的访问控制的。

数据库防火墙可以阻止来自外网对敏感数据的窃取行为，比如SQL注入攻击、后门程序等。也能阻止内网用户的越权访问和误操作。

7、数据库安全

7.2 数据库安全

7.2.3 数据库安全解决方案

- 数据库加密：

对敏感数据进行选择性加密。并建立独立于且不同于数据库的访问控制规则。

可以解决敏感数据明文存储带来的风险，也可以解决管理员权限过大，导致权利责任不统一。

- 数据库静态脱敏：

对真实数据进行定时、批量脱敏，提供准真实数据。

- 数据库动态脱敏：

内部运维人员、外包人员接触真实敏感数据，或者并且应用系统直接访问敏感数据，获取真实数据，都容易导致泄密。

动态脱敏从源头上选择性的对敏感数据进行脱敏，并控制对敏感数据的访问总量。

7、数据库安全

7.2 数据库安全

7.2.3 数据库安全解决方案

➢安全标准：

由国家公安部提出并组织制定，国家质量技术监督局发布的《计算机信息系统安全保护等级划分准则》、《信息安全等级保护管理办法》中涉及数据库安全相关规定与产品的对应关系。

等保级别	数据库审计	数据库防火墙	数据库加密	数据库脱敏
等级保护二级	**必选**（必须支持本地探针部署、数据库风险扫描）	**必选**	可选	可选
等级保护三级	**必选**	**必选**	**必选**	可选

7、数据库安全

7.2 数据库安全

7.2.4 数据库安全发展

对数据库加密是确保数据安全的重要措施。但是对数据库加密是一件困难的事情：

- 数据量大，要求加解密速度快。
- 数据存储时间长，为了安全密钥应经常更换，需要对 数据解密再加密很麻烦。
- 如不经常更换密钥，时间一长就可能不安全。
- 数据库要支持查询、插入、删除、更新等操作，最好 能在密文状态下进行上述操作，即需要同态加密

7、数据库安全

7.2 数据库安全

7.2.4 数据库安全发展

➢数据库密文检索：

对于加密后的关系数据库，其数据不再具有原本的保序性、可计算性等特点，同时关系型数据库中众多计算函数在密文环境下无法发挥作用，而计算和检索是数据库中不可或缺的核心功能。

密文检索技术能够对密文进行检索并将满足条件的数据返回给用户同时保证数据库效率，用户将密文解密后得到结果。

➢同态加密：

以企业数据库为例：假设有一个人希望找到所有员工工资的中位数。目前来看，这证明需要一个值得信赖的个人或团队可以获得员工的薪酬细节，这可能会侵犯隐私。然而，使用同态加密，可以在不解密数据，不暴露个人薪酬的情况下提取数字并得出中位数，一旦处理和解密，就只能看到最终数字。

7、数据库安全

7.2 数据库安全

7.2.5 国产数据库安全

- 国外的数据库加密产品相对较多，产品相对成熟。但面临着不能集成国产的加密算法、不符合国家安全政策，不能利用密文索引进行范围查询，造成性能严重下降等问题，因此以上产品在我国尚未得到有效应用。
- 在国内去IOE的背景下，国内专业数据库安全厂商例如安华金和、安恒、中安比特有了发展机会和市场。
- 2017年安华金和发布国内首款支持MySQL数据库的透明加密产品（DBCoffer-MySQL TDE）采用国产SM4加密算法，相较传统AES等算法更安全可靠、合规，最大程度保证数据安全。

8、大数据安全

8.1 大数据

8.1.1 定义

麦肯锡全球研究所给出的定义是：一种规模大到在获取、存储、管理、分析方面大大超出了传统数据库软件工具能力范围的数据集合

8.1.2 特点

- 海量的数据规模
- 快速的数据流转
- 多样的数据类型
- 价值密度低

8、大数据安全

8.2 大数据面临的安全问题

8.2.1 受到攻击风险高

大数据往往采用分布式方式进行存储，且终端用户数量与类型繁杂，身份认证困难，传统数据保护技术难以满足大数据安全需求，容易被攻击者绕过安全机制窃取信息。

8.2.2 隐私信息泄露风险

隐私保护技术不完善的条件下，各类软件掌握着用户的社会关系，聊天、上网、出行记录、网上支付、消费行为，人们面临的威胁不仅限于个人隐私泄露，还在于基于大数据传输对人的状态和行为的预测。如何在保证数据使用效益的同时保护个人隐私，是大数据传输时代面临的巨大挑战之一

8、大数据安全

8.2 大数据面临的安全问题

8.2.3 传输过程的安全隐患

如何在不泄露用户隐私数据的前提下进行数据挖掘是大数据安全的重要课题。在分布计算的信息传输和数据交换时，各个存储点内的用户隐私数据非法泄露和使用的风险，是当前大数据背景下信息安全的主要问题

8.2.4 大数据的存储管理风险

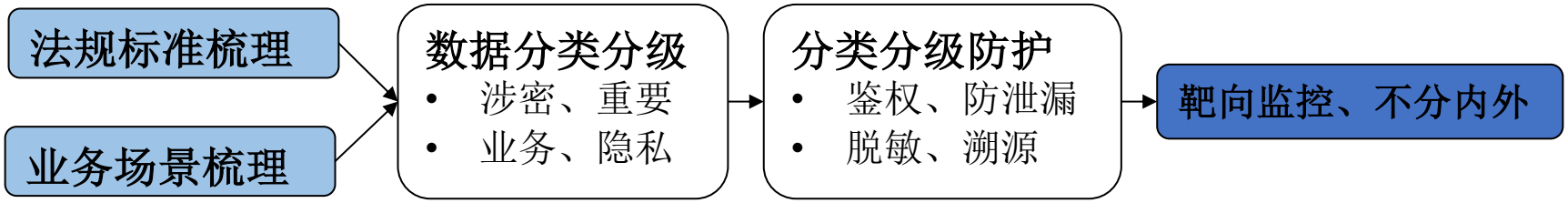
大数据存储平台，数据量是非线性甚至是指数级的速度增长的，各种类型和各种结构的数据进行数据存储，进程的并发运行管理难度极大，极易造成数据存储错位和数据管理混乱，为大数据存储和后期的处理带来安全隐患。

8、大数据安全

8.3 大数据安全治理实例

8.3.1 数据安全治理

在Gartner 2017安全与风险管理峰会上，首次提到数据安全治理（Data Scurity Governance）这一概念，定义数据安全治理绝不仅仅是一套用工具组合的产品级解决方案，而是从决策层到技术层，从管理制度到工具支撑，自上而下贯穿整个组织架构的完整链条，并认为数据安全治理是新形势下有效的数据安全防护手段。



数据安全治理思路

8、大数据安全

8.3 大数据安全治理实例

8.3.2 我国数据安全治理法律基础

《中华人民共和国数据安全法》第四条 维护数据安全，应当坚持总体国家安全观，建立健全**数据安全治理体系**，提高数据安全保障能力。

8.3.3 国内外数据安全治理参考标准

- 国际标准化组织（ISO/IEC）38505数据治理框架
- 国际数据管理协会（CDMA）DAMA-DMBOK框架
- 国际数据治理研究所（DGI）DGI数据治理框架
- IBM数据治理委员会（IBMDGA）数据治理成熟度模型
- 中国电子工业标准化技术协会信息技术服务分会（ITSS）数据治理规范

8、大数据安全

8.4 大数据安全解决方案

8.4.1 大数据安全审计

大数据平台组件行为审计，将主客体的操作行为形成详细日志、包含用户名、IP、操作、资源、访问类型、时间、授权结果、具体设计新建事件概括、风险事件、报表管理、系统维护、规则管理、日志检索等功能。

8.4.2 大数据脱敏系统

针对大数据存储数据全表或者字段进行敏感信息脱敏、启动数据脱敏不需要读取大数据组件的任何内容，只需要配置相应的脱敏策略。

8、大数据安全

8.4 大数据安全解决方案

8.4.3 大数据脆弱性检测

大数据平台组件周期性漏洞扫描和基线检测，扫描大数据平台漏洞以及基线配置安全隐患；包含风险展示、脆弱性检测、报表管理和知识库等功能模块。

8.4.4 大数据资产梳理

能够自动识别敏感数据，并对敏感数据进行分类，且启用敏感数据发现策略不会更改大数据组件的任何内容。

8.4.5 大数据应用访问控制

能够对大数据平台账户进行统一的管控和集中授权管理。为大数据平台用户和应用程序提供细粒度级的授权及访问控制。

8、大数据安全

8.5 大数据安全治理效果

8.5.1 数据安全治理预期目标

- 通过对相关法规标准的梳理，结合实际的业务场景，对信息系统中的数据进行安全性分类分级；
- 制定相应的分类分级防护策略、数据安全架构及组织制度保障，形成对数据全生命周期的安全管控；
- 对管控效果定期检查、审计、评估，建立安全组织架构和制度，并将各项安全策略与架构落实到全套安全技术手段和安全服务措施中

最终实现对敏感信息的细粒度、全方位靶向监控，建立有效的数据安全防护体系。

9 云计算安全

9.1 云计算

定义 云计算是一种可以方便地通过网络对共享的可配置计算资源池进行按需访问的模型，池中资源只需要很少的管理付出或服务提供者帮助就能快速供应和释放。

- 国内知名IT企业积极推进云平台的建设和云应用的开展

- 国外知名IT企业大力开发和推进云计算

9 云计算安全

9.2 云计算安全问题

- 产品漏洞
  - 2005 年1月，Google的Gmail因存在的安全漏洞，使用户的用户名和密码安全受到威胁。
  - 2017年2月，著名网络服务商CloudFlare爆出“云出血”漏洞。
- 隐私权限泄露
  - 2009年3月，Google发生大批用户文件外泄事件，因Google的疏忽导致用户保存在Google Docs的部分文档会在用户不知晓的情况下被共享。
  - 2017年4月，洲际酒店旗下超过1000家酒店遭遇支付卡信息泄露问题。

9 云计算安全

9.2 云计算安全问题

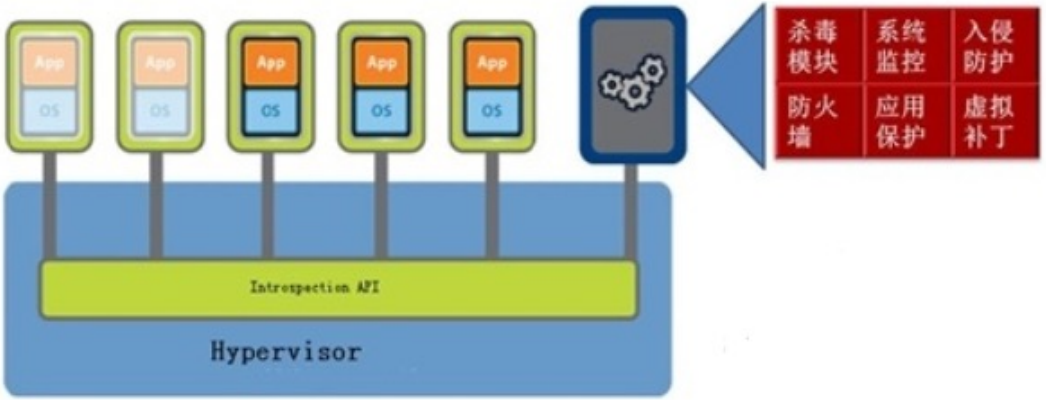
- 黑客攻击
  - 2009年，Amazon平台被僵尸网络恶意利用于非法活动。黑客通过入侵一个使用 EC2云服务的网站，在Amazon的服务器上安装了一个未授权的命令和控制程序。
  - 2017年2月，俄罗斯“黑帽”黑客获取了60多所美国大学和政府机构的访问权限。
- 服务中断
  - 2009年2月和7月，Amazon的S3服务两次中断，导致依赖于网络单一存储服务的网站瘫痪。
  - 2016年4月，Google Cloud出现18分钟服务中断。

9 云计算安全

9.5 云计算安全解决方案实例

- VMVare安全虚拟机流量清洗方案

WMWare安全保障方案思路是：在每个物理服务器内开启一个安全VM，安装安全软件，如病毒查杀。由于VM下层的Hypervisor是自己的代码，他们提供了一个API接口，通过Hypervisor控制，让进入VM的数据流先进入安全VM，安全清洗后，再流到目标VM中。实际上就是在VM的下层Hypervisor层上做每个VM的访问控制。安全VM中不仅可以安装杀病毒软件，还可以安装入侵检测、防火墙等软件，形成一个超能力的安全处理机。



9 云计算安全

9.6 云计算安全研究方向

随着云计算成为主流计算的趋势下，可信计算技术被引入云计算，用来解决云计算的安全问题，即以可信赖方式向用户提供云服务。

目前，构造可信云平台的主要方案包括：

1. 借助硬件隔离技术构造可信执行环境，如Intel的SGX、ARM的TrustZone 等，为虚拟机监控器添加安全隔离、可信验证等功能；
2. 基于可信虚拟化（如 vTPM）技术构建可信云平台；
3. 通过建立可信第三方，对云计算平台进行动态可信度量；
4. 以可信安全芯片的密钥管理为基础，将终端密钥管理转化为云平台密钥管理，借此为虚拟机提供可信服务。

10、系统安全热点问题

10.1 物联网安全

10.1.2 物联网安全概述

物联网定义 物联网是指通过各种信息传感设备，实时采集任何需要监控、连接、互动的物体或过程等各种需要的信息，与互联网结合形成的一个巨大网络，其目的是实现物与物、物与人，所有的物品与网络的连接，方便识别、管理和控制。

物联网安全定义 物联网安全指物联网硬件、软件及其系统中的数据收到保护，不受偶然的或者恶意的原因而遭到破坏、更改、泄露，物联网系统可连续可靠正常地运行。

根据其层次架构分为：感知层安全、网络层安全、应用层安全。

作为基于互联网的新兴信息技术模式，物联网安全上升至国家安全。

10、系统安全热点问题

10.1.2 物联网安全威胁

- 感知层的安全威胁
  - 物理攻击、伪造或假冒攻击、信号泄露与干扰、资源耗尽攻击、隐私泄露威胁
- 网络层的安全威胁
  - 网络层协议漏洞、海量终端设备的威胁、异构网络融合问题、无线传输问题、DDoS攻击问题
- 应用层的安全威胁
  - 病毒、蠕虫和木马
  - 不受欢迎应用程序：Rootkit、广告软件、间谍软件
  - 远程攻击：Dos攻击、DNS投毒、端口扫描、TCP去同步化、SMB中继、ICMP攻击

10、系统安全热点问题

10.1.3 物联网安全实例

10、系统安全热点问题

10.1.4 物联网安全解决策略

- 感知层的安全防护策略
  - 物理安全：设备防水、防盗、防干扰等
  - 接入安全：设备入侵防护、轻量级强制认证机制
  - 硬件安全：确保芯片内系统程序、终端参数、安全数据和用户数据不被篡改或非法获取
  - 统一安全管理：可信的身份认证、安全的固件更新等
  - 操作系统安全：行为可控
  - 应用安全：
  - 数据安全
  - DDoS攻击防护

10、系统安全热点问题

10.1.4 物联网安全解决策略

- 网络层的安全防护策略
  - 通用网络防护：包括网络结构安全，合理划分网络安全域，加强安全边界隔离，避免安全问题的扩散。
  - 网络入侵防护：部署入侵监测设备，对网络攻击进行监控和报警
  - 网络安全审计：通过统一日志管理系统或安全管理平台，对网络设备运行状况、网络流量、用户行为等进行日志审计。
  - 加密传输：在保证用户通信传输质量的同时，防止未知位置的窃听和增加中间人攻击的难度。
  - 安全路由协议

10、系统安全热点问题

10.1.4 物联网安全解决策略

- 应用层的安全防护策略
  - 身份和访问控制：应用访问时进行强制认证和业务权限控制，应尽可能采用双因素身份验证机制，加强权限管理，端口控制，敏感信息访问等。
  - 外部攻击防护：通过部署Web防火墙、IPS等设备，监控并过滤恶意的外部访问
  - APP安全：APP代码按照安全要求严格开发，做好代码加密、加壳防止反编译，APP与应用平台间数据要求加密传输，要在上线前做好评估，上线后定期评测、加固漏洞。
  - 应用安全漏洞管理：漏洞修补、漏洞检测
  - 隐私保护：常采用的安全机制包括存储加密、交换加密、身份认证与访问控制、接口安全、自我销毁技术等技术措施

10、系统安全热点问题

10.1.5 物联网安全研究方向

- 去中心化认证

区块链为物联网运行环境提供了中心化的机制，使得设备之间保持共识，无需中心验证，这样即使一个或多个节点被攻破，整体网络体系的数据依然可以可靠安全的运行。

- 边缘计算

边缘计算解决了物联网设备产生的海量数据分析和存储带来的网络带宽挑战，增强数据中心的计算能力。

10、系统安全热点问题

10.2 车联网安全

10.2.1 车联网定义

车联网是指借助新一代的移动通信技术，实现车辆内部，车与人、车与车、车与路、车与服务平台的全方位网络连接。

10.2.2 车联网安全威胁

- 网络

  网络通信安全：车辆通信过程中面临信息被窃听、篡改等风险。

  网络终端安全：云服务平台和APP由于靠近用户侧，易受网络攻击威胁。

- 平台

  车辆本身容易被攻击，例如车内CAN总线和各类传感器面临的安全威胁。

- 组件

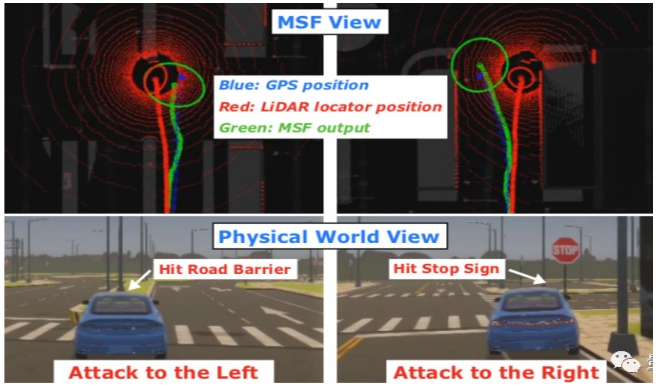
  汽车专用微机控制器（ECU）的脆弱性。

10、系统安全热点问题

10.2 车联网安全

10.2.3 安全实例

在第23届DEFCON数字安全会议上，安全专家Kevin Mahaffey和Marc Rogers演示了通过Model S存在的漏洞打开车门、启动并成功开走，此外还向Model S发送“自杀”命令,在车辆正常行驶中突然关闭系统引擎让车辆停下来。



2019年，以色列网络安全公司Regulus Cyber研究员攻击特斯拉GPS系统，使车辆驾驶辅助功能、空气悬架工作异常、车辆偏离主干道

10、系统安全热点问题

10.2 车联网安全

10.2.4 防护措施及效果

车联网防护主要分为网络、平台、组件三部分，具体包括：

- 网络级安全防护

1）通信层安全防护：
- ✓ 身份认证，对车辆接入进行身份认证。
- ✓ 入侵检测，检测车联网内的异常流量。

2）终端层安全防护：
- ✓ 云数据加密，对云平台数据进行加密存储。
- ✓ 云平台数据访问控制，对访问云平台的请求进行权限控制。
- ✓ 终端恶意代码检测，检测终端上运行代码，防范恶意代码。

效果：实现对车辆接入的高效安全认证，降低车辆被攻击风险，保障云平台和APP本身不被攻击。

10、系统安全热点问题

10.2 车联网安全

10.2.4 防护措施及效果

- 平台级安全防护

1）车内CAN总线安全防护：
- ✓CAN 总线的报文认证机制，验证报文完整性以及数据源认证。
- ✓报文加密机制，加密报文，保护其机密性。
- ✓CAN总线异常检测，检测总线信息传输异常。

2）车内传感器安全防护：
- ✓传感器状态监测，监测各类传感器运行，保证其稳定性。
- ✓传感器攻击检测，检测异常攻击行为。

效果：通过保障车内CAN总线和传感器的安全，保障车辆自身的安全性。

10、系统安全热点问题

10.2 车联网安全

10.2.4 防护措施及效果

➢组件级安全防护

1）漏洞检测：

✓基于静态分析的漏洞检测，对程序源代码进行对比分析，定位存在漏洞。

✓基于符号执行的漏洞检测，利用符号执行引擎，分析代码执行路径，从而定位漏洞。

✓基于模糊测试的漏洞检测，向被测程序输入半随机数据并执行程序，分析程序发生的异常，从而发现漏洞。

2）固件保护：

✓固件反逆向，给固件代码中插入混淆指令，避免使用开源指令集。

✓固件安全升级，从官方源更新固件。

效果：通过对组件（例如ECU）源代码进行漏洞检测，防范漏洞风险，并进行安全的固件更新，保证固件的安全，从而保证组件层次的安全。

10、系统安全热点问题

10.2 车联网安全

10.2.5 国内主要研究问题

➢如何提高针对车联网攻击的检测及攻击响应效率？

目前针对车联网的恶意攻击检测不及时，响应慢，如何提高检测和攻击响应的效率、准确性是目前研究的重点之一。

➢如何实现精准的网络安全测试评估？

由于未来智能网联车中使用的异构软件和硬件组件的复杂性日益增加，如何对这些新技术的集合体进行安全测试评估也是目前还未解决的问题。

10、系统安全热点问题

10.2 车联网安全

10.2.5 国内主要研究问题

- 如何应对未知的智能网联车网络攻击？

车联网环境变化性强，攻击类型多样，未知攻击也不断出现，如何应对未知攻击也是急需解决的问题之一。

- 如何制定及时有效的智能网联车信息安全标准？

标准化建设是有效提高汽车产品开发协同效率、降低开发和维护成本的有效措施。目前，国内的标准存在一定的滞后现象，因此如何制定及时有效的车联网信息安全标准是未来需要解决的重要问题。

10、系统安全热点问题

10.3 空天信息安全

空天信息安全包括着终端安全、环境安全与网络安全等多个方面。

➢具体案例

21世纪初我国鑫诺卫星遭受攻击，攻击者通过发送与正常信号频谱特征相近的非法干扰信号，造成正常电视信号无法传播。

2015年日本“瞳”卫星因姿势控制系统故障，而地面控制端输入指令错误，卫星没有故障与错误指令预防机制，进而造成卫星无法正常运行并坠毁。

10、系统安全热点问题

10.3 空天信息安全

- 卫星链路安全问题
- 可信的空天信息
- 传统的安全问题与非传统环境和应用的交织

总结

- 系统安全是网络安全领域最基本的问题之一
- 系统安全问题覆盖范围广，内容繁多
- 细分领域和问题不断的发展和延伸
- 哪里有系统，哪里就有安全问题

- 自主、可信是解决和改善系统安全问题的最终道路

End

谢　谢！