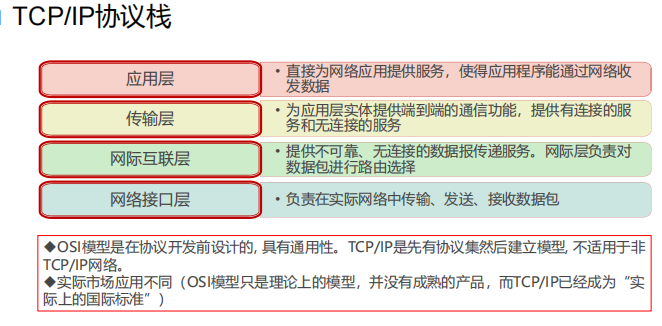
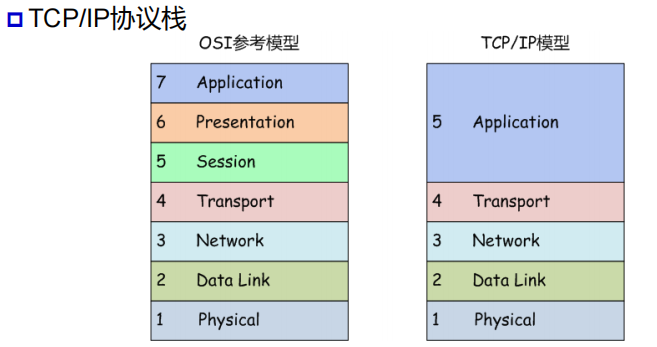
1. **网络攻防的基本概念** 1.1 **网络安全的属性** **网络空间安全**(Cybersecurity)： 研究网络空间中的安全威胁和防护问题。包括网络空间基础设施的安全和可信，以及网络空间信息的保密性、完整性、可用性、真实性和可控性的相关理论与技术。 **核心属性**（CIA）： **保密性**(Confidentiality)：确保隐私或者秘密信息不向非授权者泄漏，也不被非授权者使用，即：防止数据的未授权访问。 **完整性**（Integrity）：确保信息只能以特定和授权的方式进行改变，比如：确保接收者收到的消息就是发送者发送的消息。 **数据完整性**：确保信息和程序仅以授权方式进行变动 **系统完整性**：确保系统以非损害方式完成设计的功能，免受有意或者无意对系统的非授权操作。 **可用性**（Availability）：合法用户在需要使用网络资源的时候，能够获得正常的服务。 **可控性**：限制对网络资源（软件和硬件）和数据（存储和通信的数据）的访问，其目标是防止未授权使用资源、未授权公开或者修改数据。通过访问控制实现。 **不可否认性**（Non-reputation）：通信实体不能对自己做过的事情抵赖，包括两层含义，一方面发送者不能否认自己发送数据的行为；另一方面，接收者不能否认自己接收过数据。 **真实性**（authenticity）：一个实体是其所声称实体的这种特性。 **可靠性**（reliability）：与预期行为和结果一致的特性。 1.2 **典型网络安全事件：** **Morris worm（蠕虫）、Mirai botnet（DDos）、Stuxnet（震网蠕虫病毒）、WannaCry Ransomware（勒索病毒）、FireEye Redteam tools（泄露事件）、 Solarwinds（供应链攻击事件）** **病毒、蠕虫、木马：**病毒、蠕虫和木马是可导致计算机和计算机上的信息损坏的恶意程序。这三种东西都是人为编制出的恶意代码，都会对用户造成危害。 **计算机病毒**(Computer Virus)，根据《中华人民共和国计算机信息系统安全保护条例》，病毒的明确定义是“指编制或者在计算机程序中插入的破坏计算机功能或者破坏数据，影响计算机使用并且能够自我复制的一组计算机指令或者程序代码”。病毒往往具有很强的感染性、潜伏性、特定的触发性、破坏性。 **病毒必须满足两个条件：自行执行**：它通常将自己的代码置于另一个程序的执行路径中。**自我复制：**例如，它可能用受病毒感染的文件副本替换其他可执行文件。病毒既可以感染个人计算机也可以感染网络服务器。 **蠕虫(worm)病毒**是一种常见的计算机病毒，它利用网络（网络、电子邮件等）进行复制和传播。蠕虫病毒是自包含的程序（或是一套程序），它能传播自身功能的拷贝或自身的某些部分到其他的计算机系统中（通常是经过网络连接）。（典型蠕虫病毒：震网病毒、勒索病毒） **普通病毒与蠕虫病毒的区别**：复制方式上，普通病毒需要传播受感染的驻留文件来进行复制，而蠕虫不使用驻留文件即可在系统之间进行自我复制；传染目标上，普通病毒的传染能力主要是针对计算机内的文件系统而言，而蠕虫病毒的传染目标是互联网内的所有计算机。 **木马(Trojan Horse)，**是指那些表面上是有用的软件、实际目的却是危害计算机安全并导致严重破坏的计算机程序。（是一种用于远程控制的黑客工具、具有隐蔽性、具有非授权性） **隐蔽性**是指木马的设计者为了防止木马被发现，会采用多种手段隐藏木马，这样服务端即使发现感染了木马，也难以确定其具体位置。**非授权性**是指一旦控制端与服务端连接后，控制端将窃取到服务端的很多操作权限，如修改文件、修改注册表、控制鼠标、键盘、窃取信息等等。 **木马与病毒的区别**：木马不具传染性，它并不能像病毒那样复制自身，也并不“刻意”地去感染其他文件，它主要通过将自身伪装起来，吸引用户下载执行。木马一般主要以窃取用户相关信息或隐蔽性控制为主要目的，相对病毒而言，可以简单地说，病毒破坏你的信息，而木马窥视你。 **软件漏洞** **网络系统的脆弱性：**指系统的硬件资源、通信资源、软件及信息资源等存在的弱点和缺陷。包括硬件系统的弱点和缺陷、软件系统的弱点和缺陷、网络和通信协议的弱点和缺陷、使用者的弱点 **软件安全漏洞**：一台计算机是由硬件以及软件两个部分组成，最基本的软件就是操作系统。 计算机软件是由计算机程序员开发出来的。不同程序员的编程水平不一样，就会造成软件存在这样或者那样的问题，这些问题可能会造成软件崩溃不能运行，我们称这些问题为**软件缺陷（Bug）** 软件中存在的一些问题可以在某种情况下被利用来对用户造成恶意攻击，如给用户计算机上安装木马病毒，或者直接盗取用户计算机上的秘密信息，等等。这个时候，软件的这些问题就不再只是Bug，而是一个软件安全漏洞，简称**“软件漏洞”** 电脑肉鸡：也就是受别人控制的远程电脑。肉鸡可以是各种系统,如windows,linux,unix等；更可以是一家公司、企业、学校甚至是政府军队的服务器。如果服务器软件存在安全漏洞，攻击者可以发起“主动”进攻，植入木马，将该服务器变为一个任人宰割的“肉鸡” **漏洞产生的原因** 1. 小作坊式的软件开发：质量参差不齐 2. 赶进度带来的弊端：投机取巧或者省工省料的办法来开发软件3. 被轻视的软件安全测试：功能为上，测试为下4. 淡薄的安全思想：缺乏安全开发的意识和经验 5. 不完善的安全维护：不重视安全维护，不重视漏洞修复 **漏洞分类：**一个漏洞从被攻击者发现并利用，到被厂商截获并发布补丁，再到补丁被大多数用户安装导致漏洞失去了利用价值，一般都要经历一个完整的生命周期。按照漏洞生命周 期的阶段进行分类的方法包括三种： **0-day漏洞：**指还处于未公开状态的漏洞。这类漏洞只在攻击者个人或者小范围黑客团体内使用，网络用户和厂商都不知情，因此没有任何防范手段，危害非常大。（0-day漏洞也是当前网络战中的核武器） **1-day漏洞：**原义是指补丁发布在1天内的漏洞，不过通常指发布补丁时间不长的漏洞。由于了解此漏洞并且安装补丁的人还不多，这种漏洞仍然存在一定的危害。 **n-day漏洞/已公开漏洞：**已公开漏洞是指厂商已经发布补丁或修补方法，大多数用户都已打过补丁的漏洞。这类漏洞从技术上因为已经有防范手段，并且大部分用户已经进行了修补，危害比较小。 电脑肉鸡是被别人控制的远程电脑。将大量服务器沦为肉鸡，主要依赖于**软件漏洞、木马**。 **漏洞库：**大量软件漏洞需要一个统一的命名和管理规范，以便开展针对软件漏洞的研究，提升漏洞的检测水平，并为软件使用者和厂商提供有关软件漏洞的确切信息。多个机构和相关国家建立了漏洞数据库，这些数据库分为公开的和某些组织机构私有的不公开数据库。公开的数据库包括CVE、NVD、BugTraq、CNNVD、CNVD等。 通过漏洞信息数据库，可以找到操作系统和应用程序的特定版本所包含的漏洞信息，甚至针对某些漏洞的专家建议、修复办法和专门的补丁程序。极少的漏洞库还提供检测、测试漏洞的POC。POC（proof-of-concepts，为观点提供证据）: 样本验证代码。 目前，许多国家建立了针对漏洞的应急响应机构，例如美国计算机应急反应小组US-CERT，中国的国家互联网应急中心CNCERT/CC。他们是软件漏洞数据的主要提供者或者漏洞库的主要维护者，并且提供了高风险的漏洞警报和专家建议。 **美国国家漏洞数据库NVD**（National Vulnerabilities Database）同时收录三个漏洞数据库的信息，CVE漏洞公告、US-CERT漏洞公告、USCERT安全警告，也自己发布漏洞公告和安全警告，是目前世界上数据量最大，条目最多的漏洞数据库之一。 **（美国）通用漏洞列表 CVE(Common Vulnerabilities andExposures)相当于软件漏洞的一个行业标准。**它**实现了安全漏洞命名机制的规范化和标准化**，为每个漏洞确定了唯一的名称和标准化的描述，为不同漏洞库之间的信息录入及数据交换提供了统一的标识，使不同的漏洞库和安全工具更容易共享数据，成为评价相应入侵检测和漏洞扫描等工具和数据库的基准。 **中国国家信息安全漏洞库CNNVD（**China National Vulnerability Database of Information Security）隶属于中国信息安全测评中心，是中国信息安全测评中心为切实履行漏洞分析和风险评估的职能，负责建设运维的国家级信息安全漏洞库，为我国信息安全保障提供基础服务。 **国家信息安全漏洞共享平台CNVD**（China National VulnerabilityDatabase）是由CNCERT/CC（国家计算机网络应急技术处理协调中心）联合国内重要信息系统单位、基础电信运营商、网络安全厂商建立的信息安全漏洞信息共享知识库，致力于建立国家统一的信息安全漏洞收集、发布、验证、分析等应急处理体系。 **其他**：EDB漏洞库、微软安全公告板和微软安全建议、绿盟科技的中文安全漏洞库、启明星辰的中文安全公告库 **术语解释**：YARA：一款VirusTotal开发的用于恶意软件识别和分类的规则匹配工具。；Snort：一款历史悠久的开源网络入侵检测系统。； IOC：失陷指标（Indicator Of Compromise），即攻击者控制被害主机所使用的远程命令及控制服务器的相关情报。IOC往往是域名、IP、URL等，这种IOC可部署于安全设备（如：IPS（入侵防御）、SIEM（安全、信息和事件管理）等）进行检测发现甚至实时阻截。； ClamAV：Linux平台上的开源病毒扫描程序，主要应用于邮件服务器，采用多线程后台操作，可以自动升级病毒库。 **安全威胁与安全攻击** **安全威胁**：威胁是指对安全可能的侵害，这种侵害并不意味着实际发生。正因为这种侵害可能发生，因此需要进行防范。如果这种侵害发生了，则称为攻击，而执行攻击行为的主体则称为攻击者。 威胁的主要类型： 信息泄露 、完整性破坏、服务拒绝、未授权访问 **威胁来源**：内部操作不当、内部管理漏洞、外部安全威胁。 **安全攻击**：任何危及到信息安全的行为（安全攻击往往要利用一个或多个威胁） 安全攻击的类型（IATF定义）：被动攻击、主动攻击、物理临近攻击、内部人员攻击、配装分发攻击。 **被动攻击**：攻击者只是窃听消息，不对消息做任何形式的修改。攻击者的目标是获取传输的信息，以便进行利用。 被动攻击的后果：信息内容泄漏、流量模式泄漏 对策：流量加密、流量混淆 常用软件 Wireshark、 Sniffer Pro、TCP Dump、 Snort 典型被动攻击：国家级监控：“棱镜”计划（PRISM） **主动攻击**：避开或破坏安全部件， 引入恶意代码，破坏数据或系统完整性。 如：假冒某个实体主动发送消息、重放旧消息（re-play）、修改传输中的消息、删除选中的消息、各种MiTM攻击。 对策：增强区域边界保护、基于网络管理交互身份认证的访问控制、受保护远程访问、质量安全管理、自动病毒检测工具、审计和入侵检测。 主动攻击特点--可以检测：由于构成系统的物理通信设施、软件和网络协议等存在各种潜在的弱点，因此主动攻击难以绝对防御，但是可以检测。因此，针对主动攻击，重点在于检测并从破坏中恢复过来。 **物理临近攻击** ：一个未授权的个人近距离物理接触网络、系统或设备，以修改、收集信息或者拒绝对信息的访问。这种接近可以通过秘密进入、公开访问或者两者结合。 对策：配置环境监控体系，提供设备物理安全保护 **内部人员攻击**：由在信息安全处理系统物理边界内的合法人员或者能够直接访问信息安全处理系统的人员发起的攻击。 对策：安全意识和训练；审计和入侵检测；安全策略和增强安全性；关键数据、服务和局域网的特殊的访问控制；加强身份识别与认证能力等。 **配装分发攻击**：硬件或软件在生产与安装过程中，或者在运输过程中，被恶意地修改。 对策：可以通过加强处理配置控制将这类威胁降低到最低。通过使用受控分发，或使用由最终用户检验的签名软件和存取控制可以降低分发威胁。 1.3 **网络攻击的策略** **APT攻击（Advanced Persistent Threat）**是指一种隐秘而持久的网络攻击，攻击组织通常由一个国家资助，具有政治、军事以及经济动机，通过网络攻击手段获得高价值目标网络的访问权限，并在目标系统中维持较长时间。 APT行动者的目的是获得并渗出高度机密的信息，如专有技术的信息，如F22的源代码等；破坏目标系统的资源完整性，如stuxnet中的PLC等。 **Lockheed Martin：Cyber Kill chain（攻击链）：** 侦察（Reconnaissance, Recon): 攻击者选择目标，研究目标，试图识别目标网络的漏洞。 武器研制（Weaponization): 攻击者针对目标情况研制远控恶意软件武器，如：病毒、蠕虫，或者恶意PDF文档、恶意office文档等。 投递（Delivery）: 攻击者传输攻击武器到目标，比如：通过电子邮件附件，网站，或者u盘 利用（Exploitation）: 恶意软件被触发，造成目标系统的漏洞被利用。 安装（Installation）: 恶意软件安装可以被攻击者使用的访问点（如：后门）以获得持久访问。 命令与控制（Command & Control）: 恶意软件使得攻击者能够持久访问目标网络。 目标行动（Action on Objective）: 攻击者采取行动来达到其目标，如：数据外泄，数据破坏，加密勒索，入侵其它目标 **Mitre：ATT&CK框架** Mitre提出了ATT&CK框架，全称为敌手策略、技术与通用知识Adversarial Tactics, Techniques, and Common Knowledge），是一个基于对真实世界观察而总结出的敌手策略与技术知识库。 策略（Tactic）是指敌手的技术目标（the adversary’s technical goal） 技术（Technique）是指敌手如何实现技术目标 过程（Procedure）是指技术的具体实现 ATT&CK以矩阵形式呈现，目前分为三类，分别为：Enterprise、Mobile和ICS，每个矩阵还可以进一步细化为更具体的多个矩阵。在此基础上，还增加了一个PRE-ATT&CK矩阵，用以完善整个知识库。 **企业矩阵** 表示敌手针对企业信息系统的攻击策略与技术，主要包含下述平台的信息：Windows, macOS, Linux, AWS, GCP, Azure, Azure AD, Office 365, SaaS. 该矩阵的部分截图如下，共包含12项策略，每项策略有多种实施技术，每项技术又对应多种具体实现。 **PRE-ATT&CK矩阵** 企业矩阵起始于初始访问，但是无论从攻击还是防御来说都是不完善的。 如果按照企业矩阵进行防御，则企业可以采用周界防御措施（如防火墙）根据IOC（Indicator of compromise）（如：IP地址、域名、恶意软件哈希值等等）制定的黑名单进行封堵，然而这种防御方式的效果往往是有限的。比如Verizon曾经报道过99%恶意软件的哈希值的可见时间仅小于或等于58秒，显然仅仅采用周界防御措施是有局限性的。 仅用企业矩阵是不足以制定完善的防御规划，防御方不仅需要监视和理解敌手在企业周界之内的活动，而且也需要将这些操作扩展到企业之外。PRE-ATT&CK用于描述敌手发起攻击之前的活动。 1.4 **构建安全系统的原则** **最小权限原则（least priviledge)、默认故障安全原则（fail-safe defaults）、安全机制的经济性原则（economy of mechanism）、完全仲裁原则（complete mediation）、开放设计原则（open design)、权限分离原则（separation of priviledge）、最少共用机制（least common mechanism)、心理可接受原则（psychonogical acceptability）、纵深防御原则（defense in depth）** **最小权限原则**：主体（用户、程序等）应该仅被授予完成任务所需的访问权限。主体拥有的权限越少，在发生安全问题（如用户登录凭证泄露）时造成的损失越小。（例如：一个操作系统，所有用户都拥有全部权限，那么任何一个用户的登录凭证泄露都会造成整个系统沦陷。反之，如果按照最小权限原则来分配权限，则某个用户的登录凭证泄露所带来的风险仅限于该用户的权限范围） **默认故障安全原则：**如果一个主体没有被明确授权访问一个对象，该主体应该被拒绝访问该对象。默认 故障安全原则要求对一个对象的默认访问是无权限。任何时候，只要安全相关的属性（如访问权限等）没有明确授予，则应被拒绝。而且，如果主体不能在对象上完成其操作，则主体应该在终止之前取消对系统安全状态的改变。该原则用于限制在主体或者对象创建时，如何初始化权限。（例如：如果邮件服务器由于配额等原因不能向指定目录写入邮件消息，则该邮件服务器也不能向其它地方写入邮件数据，而应该关闭网络连接并报告错误。如果该邮件服务器能够向其它地方写入邮件数据，则可能被攻击者利用，通过发送大量邮件来填满其它存储空间，可能导致整个系统崩溃）。 **安全机制的经济性原则：**安全机制的经济性原则要求“安全机制应该尽可能简单”如果设计比较简单，则机制的构件比较少，从而实现容易，所需的测试用例相对较少，安全机制出错的概率大大降低。 **完全仲裁原则**：对资源的所有访问均需要审核。任何时候，当一个主体试图读取一个对象，操作系统都应该审核该行为。首先，操作系统应该确定主体是否被许可读取该对象；如果是，那么允许当前的读操作。之后，如果主体再次试图读取该对象，操作系统应该再次审核该操作是否被允许。 **开放设计原则：**和Kerckho**ff原则一致，密码系**统应该在就算攻击者知道所有系统内部细节的情况下也保持安全。对于密码系统来说，密钥应该是唯一需要保密的，系统应该被设计为容易更换密钥，因此在密钥泄露的情况下，通过更换密钥，系统仍然能够保持安全性。典型实例：IKE、IPSec、TLS、WPA、RSA **权限分离原则**： 系统不能基于单一条件来授予访问许可，或者说系统需要同时满足多个条件时，才授予某个主体对某个资源的访问许可。（例如：在BSD unix中，普通用户可以使用su命令切换到root用户，但是需要两个条件，该用户知道root的口令，用户的组ID为0。双因素身份认证，如银行卡取钱，需要提供银行卡和PIN；某些重要网站的登录需要提供账号，移动认证码） **最少共用机制**：互不信任的用户之间访问资源的共用机制应该最小化，包括：共用子系统、共享资源、共享代码等等。因为共用机制提供了一种在攻击者和被攻击者之间的潜在通道，攻击者可能利用这种通道进行攻击。（例如：虚拟机和容器，虚拟机之间共享的资源少于同一平台上容器之间共用的资源，而事实上也证明，在隔离效果上虚拟机比容器更安全。DNS缓存毒化攻击，其能够成功的原因也在于用户之间共用了DNS缓存。） **心理可接受原则**：安全机制不应该增加访问资源的难度，安全机制应该易于使用。在引入安全机制时，需要考虑如何屏蔽安全机制的复杂性，达到易于安装、配置和使用。如果做不到心理可接受性，用户可能会采取措施导致安全机制失效。（例如：安全管理员要求员工的口令必须是随机生成且包含大小写字母、数字、特殊符号，且不低于16个字符，且每周更新。这种要求通常会导致员工把口令写到便利贴上，贴到容易看到的地方，这种方式本质上导致了安全机制失效。） **纵深防御**：指在一个信息系统（包括：单机系统、网络系统等）中部署多层次的安全控制（防御）措施，其目的是提供冗余的安全保护，同时不同防御措施之间还存在互补性，以免一个防御措施失效，整个系统沦陷。（比如，在我们的个人电脑上通常都部署了防火墙和防病毒软件，而高价值的服务器还会部署HIDS等。在企业网络环境，还存在更复杂的各种安全措施） 2. **互联网协议的安全性分析**

**1.TCP/IP协议安全性概述**



**物理层（Physical Layer）**： 负责在物理媒介上传输数据比特流，包括传输介质、电压等物理特性。

**数据链路层（Data Link Layer）**： 提供可靠的点对点数据传输，负责数据帧的传输、错误检测和纠正，以及物理地址寻址。

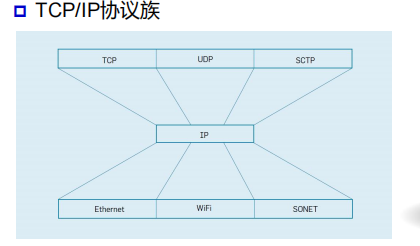
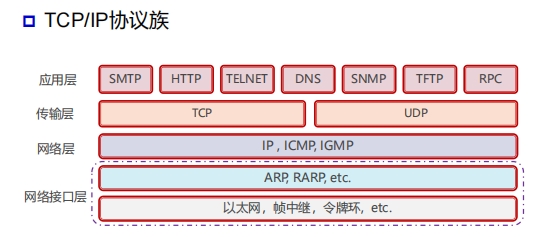
**网络层（Network Layer）**： 负责在网络上选择路由并传输数据包，处理逻辑地址（IP地址），实现数据分组的路由和转发。

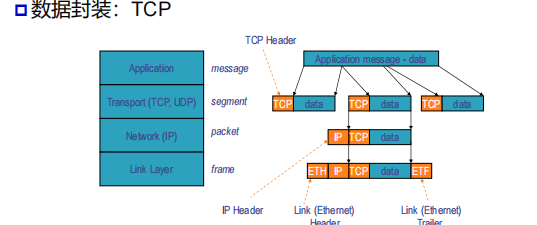
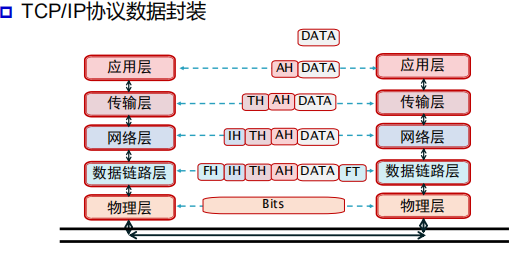
**传输层（Transport Layer）**： 提供端到端的数据传输服务，确保可靠的数据传输，包括数据分段、错误校验、流量控制等。

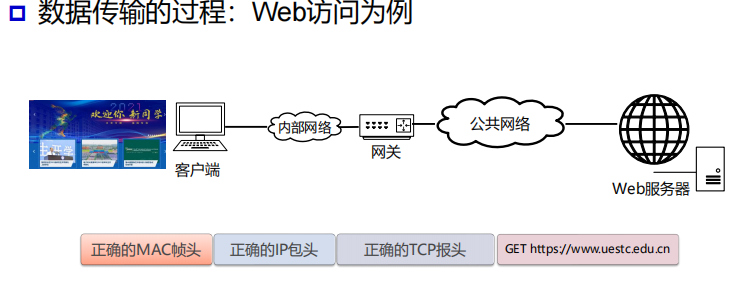
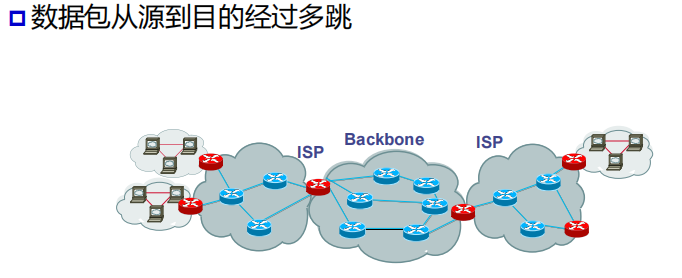
会话层（Session Layer）： 管理不同计算机之间的会话，包括建立、管理和终止通信会话。

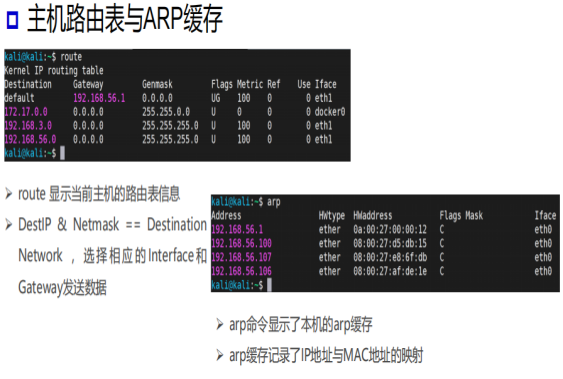
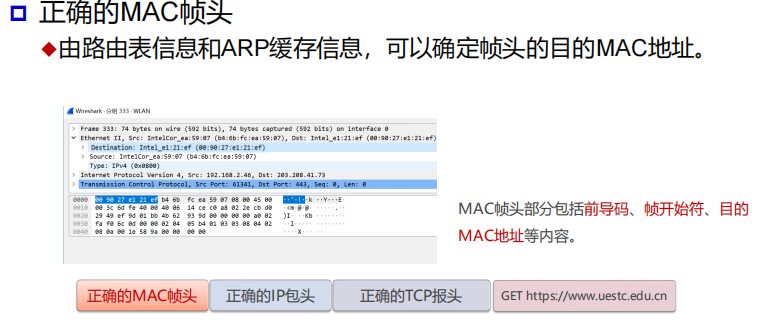
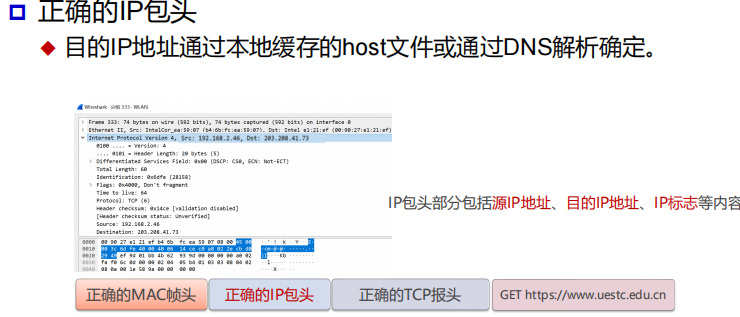
表示层（Presentation Layer）： 负责数据格式的转换和加密解密，确保不同系统之间的数据格式兼容性。

**应用层（Application Layer）**： 提供用户接口和应用程序之间的通信服务，包括识别通信伙伴、数据交换和网络服务。









**安全问题**的来源：互联网设计之初的使用目的是用于科学研究，其基本假设就是节点的诚实性；由于计算机网络的广泛使用，这种假设在今天已经无法成立，因此可能导致各种各样的攻击。

安全性问题分类：

**设计缺陷导致的安全性问题**：协议设计的缺陷，这类安全性问题会一直存在，直至该协议更新

**实现缺陷导致的安全性问题**：协议实现的缺陷，这类安全性问题会随着软件的更新而消除

**信息泄露**：TCP/IP协议在设计时没有考虑保密性服务，所有消息均通过明文方式传输，导致消息在传输过程中存在信息泄露的安全威胁。

**消息伪造**：TCP/IP协议在设计时没有考虑身份认证和完整性服务，导致消息容易被伪造。

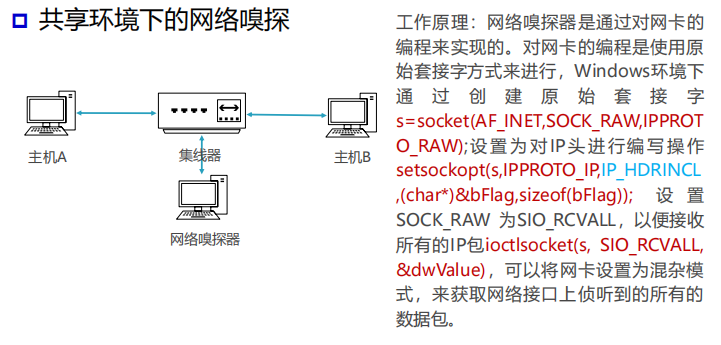
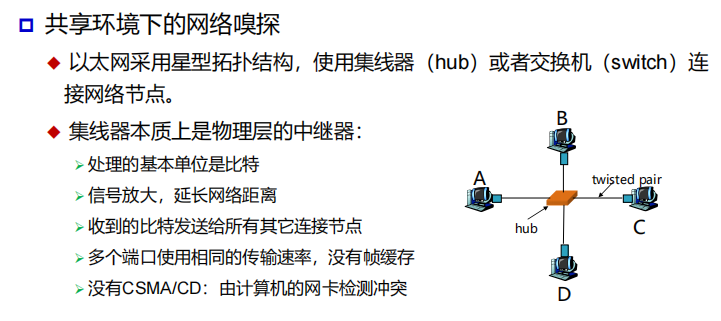
**拒绝服务**：TCP/IP协议在设计时没有考虑可用性服务，导致拒绝服务攻击。

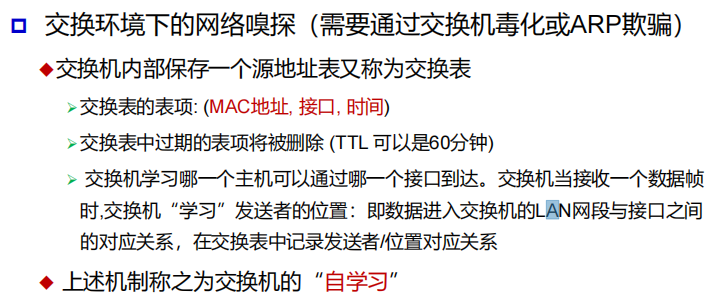
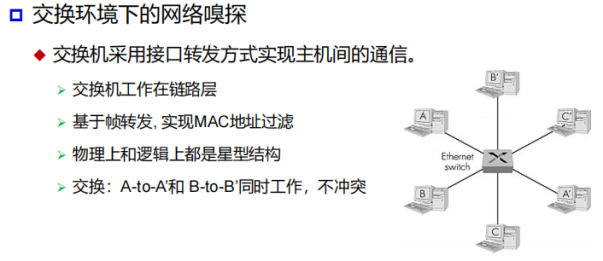
安全威胁分布： 应用层• DNS欺骗，邮件炸弹 传输层• SYN Flood攻击，会话挟持

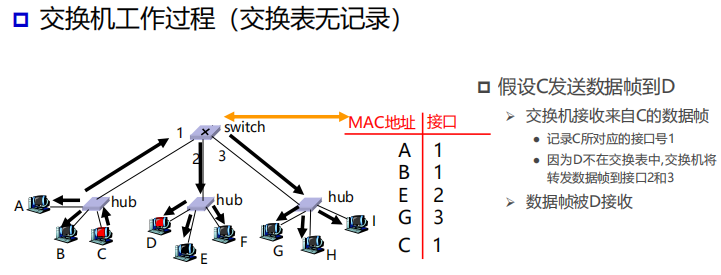
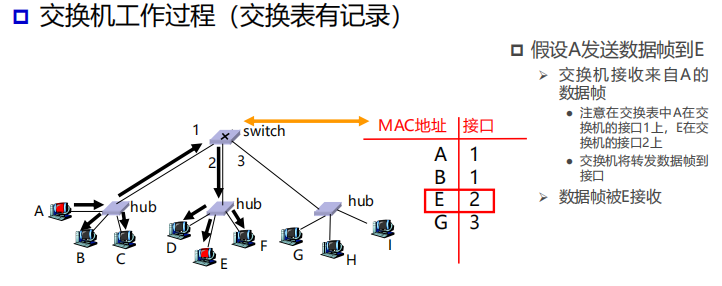
网络层• ICMP重定向攻击，IP分片攻击 网络接口层• 嗅探，ARP欺骗，交换机毒化

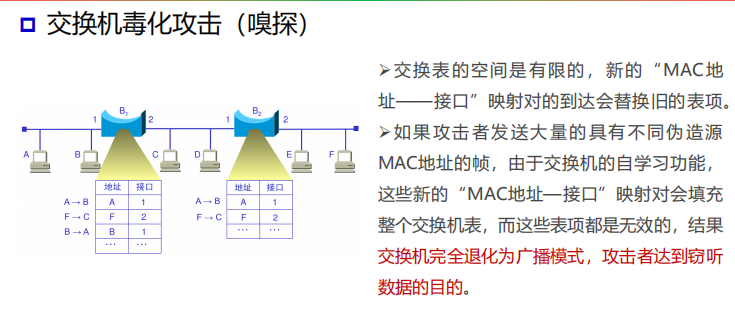
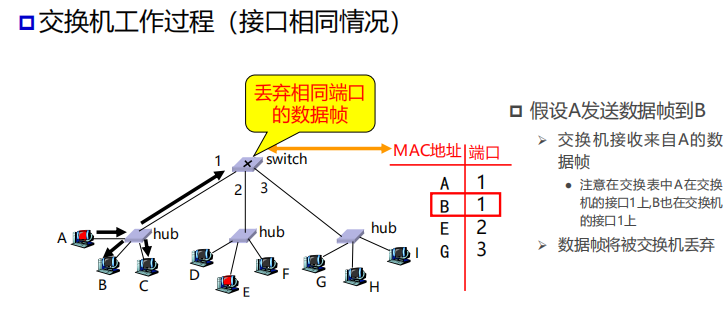
**2 网络接口层协议安全分析**

**网络嗅探**

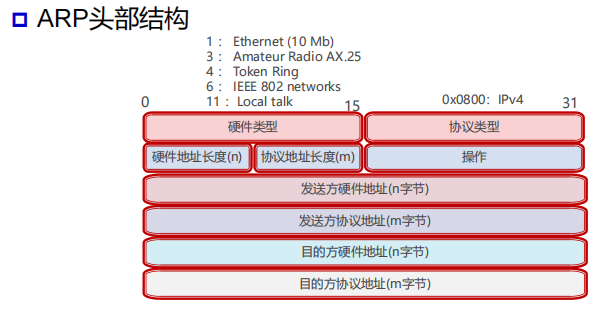
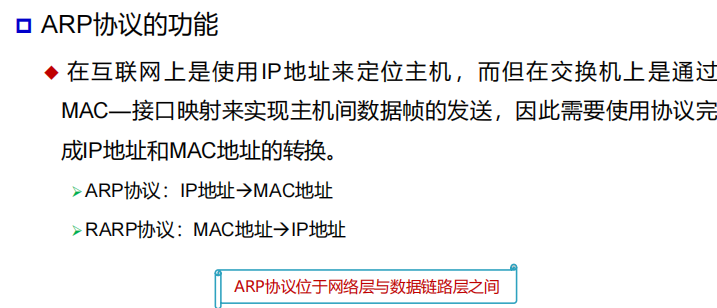


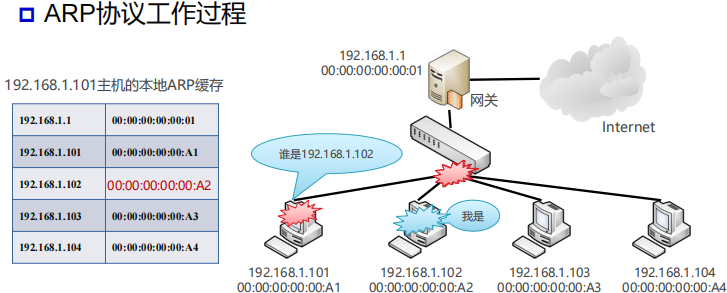
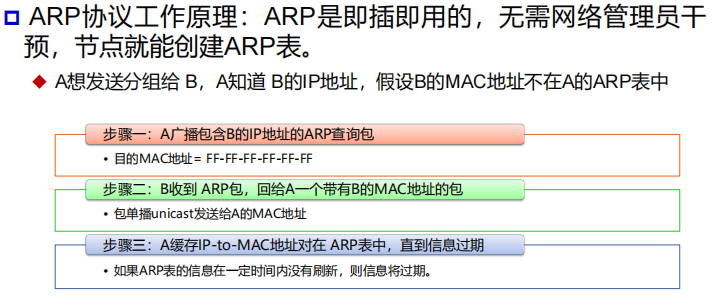






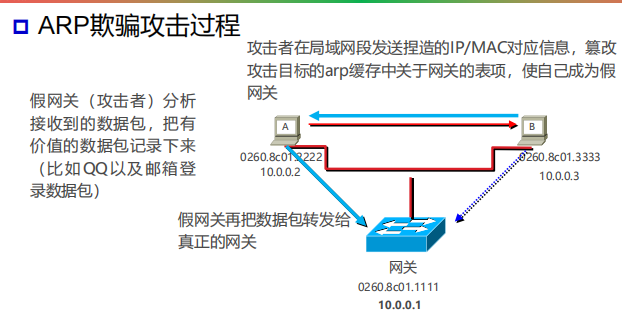
**ARP欺骗**





**ARP协议的特殊设计**（改进效率）：响应ARP请求的主机将请求者的IP－MAC映射缓存；主动的ARP应答会被视为有效信息接受

**ARP协议的缺陷**：ARP协议设计之初没有考虑认证问题，所以任何计算机都可以发送虚假的ARP数据包；ARP协议的无状态性。响应数据包和请求数据包之间没有什么关系，如果主机收到一个ARP响应却无法知道是否真的发送过对应的ARP请求；ARP缓存需要定时更新，给攻击者以可乘之机



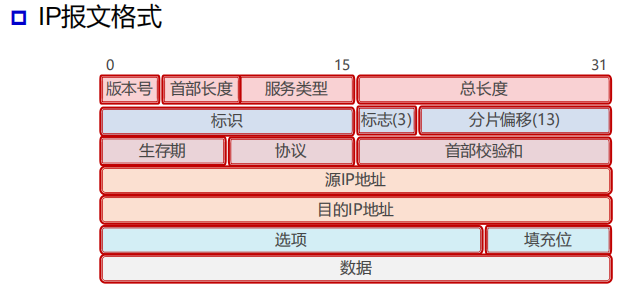
ARP欺骗攻击的特点

**危害**：嗅探、中间人攻击、拒绝服务攻击

**局限性**：ARP欺骗只能被用于局域网（攻击者必须已经获得局域网中某台机器的访问权）

**3 IP协议安全性分析**

**IP假冒攻击**

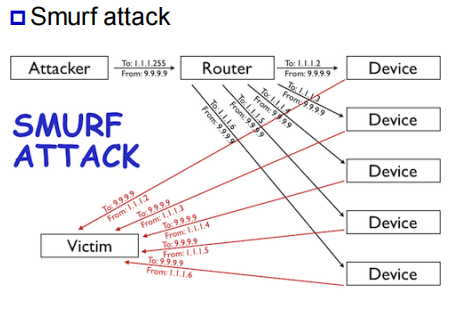


**攻击原理**：IP协议本身没有验证源IP地址真实性的机制

**攻击类型**：

拒绝服务--避免被追踪而受到惩罚，构造针对同一目的 IP 地址的 IP 分组，而源 IP 地址为随机的IP地址。

基于 IP 地址认证的网络服务欺骗--假冒可信的IP 地址而非法访问计算机资源，X-window、rlogin、rsh等。



步骤：

1、攻击者构造ICMP报文，源IP地址为受害者的IP地址，目的IP地址为一个网络的广播地址。然后，发送出去。

2、目标网络的主机收到ICMP报文后，以受害者的IP地址为目的IP地址进行响应。

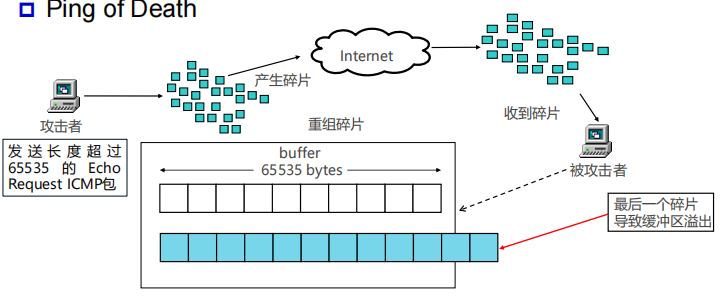
3、结果，受害者被DDos。

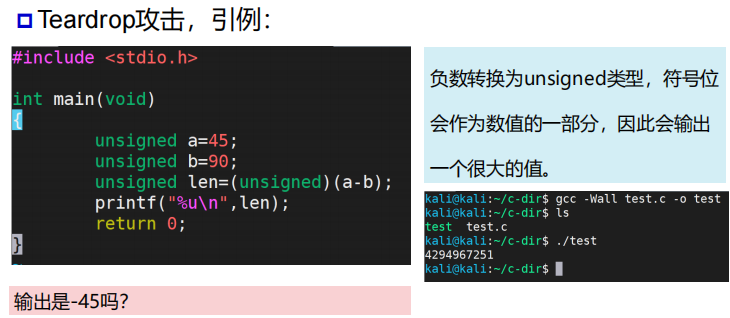
**IP碎片攻击**

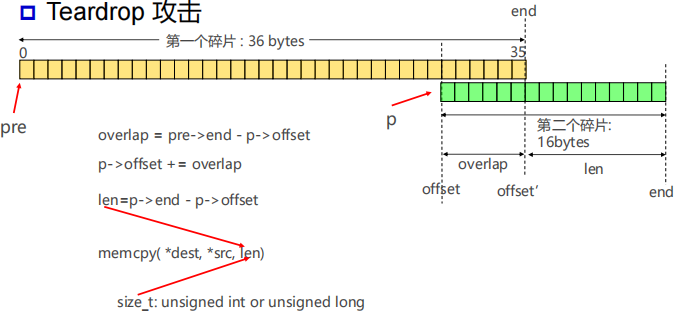
攻击原理

链路层具有最大传输单元MTU这个特性，它限制了数据帧的最大长度，不同的网络类型都有一个上限值，以太网的MTU是1500。如果IP层有数据包要传，而且数据包的长度超过了MTU，那么IP层就要对数据包进行分片(fragmentation)操作，使每一片的长度都小于或等于MTU。

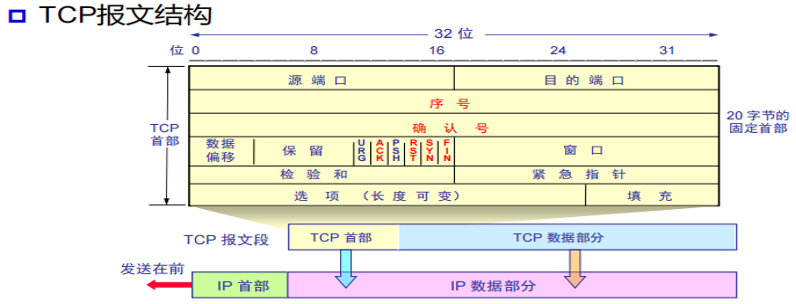
IP首部有两个字节表示整个IP数据包的长度，所以IP数据包最长只能为0xFFFF，就是65535字节。如果有意发送总长度超过65535 的IP报文，或构造畸形的IP碎片，部分老的操作系统在进行碎片重组处理时会导致系统崩溃或拒绝服务。

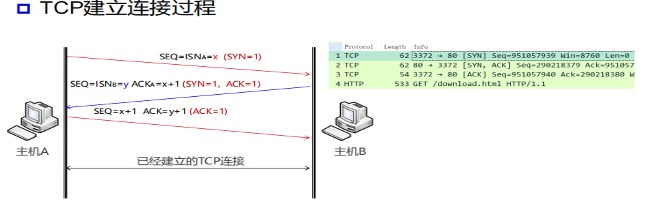


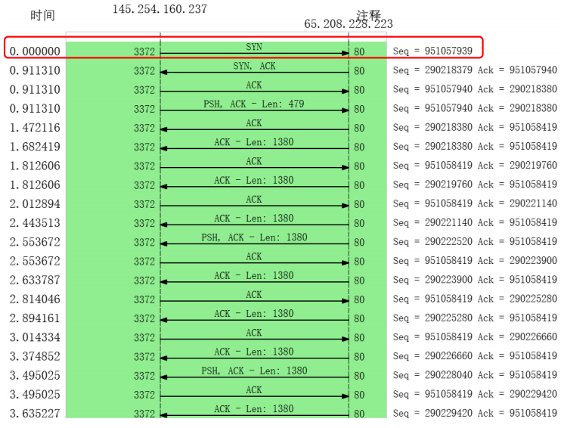




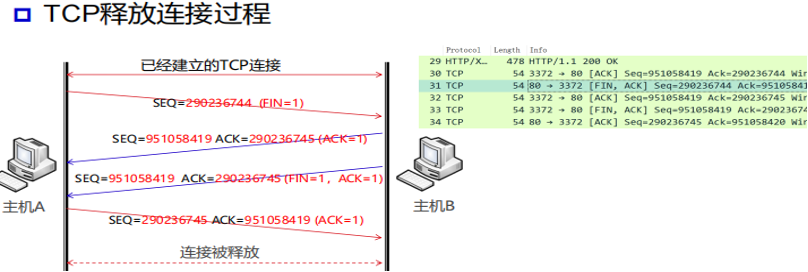
2.4 **传输层协议安全性分析** **TCP协议安全威胁**



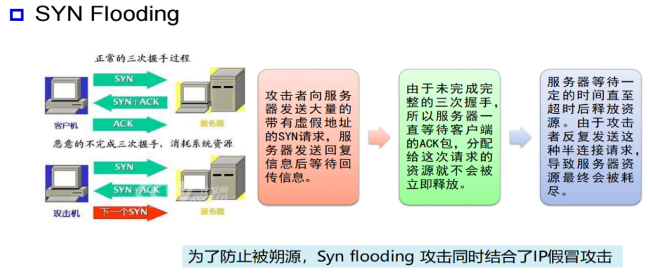




**序列号和确认号**（SYN用于初始化连接，而ACK则用于确认收到数据或者确认连接的建立）**包1**：客户端发起建立连接请求，SYN标志为1，序 号 为 随 机 生 成 的 整 数951057939 **包2**：服务端响应客户端的建立连接请求，SYN和ACK标志均置为1，序号为随机生成的整数290218379，确认号为客户端的初始序号+1即：951057939 + 1 = 951057940 需要注意的是，尽管客户端没有发送任何有效数据，确认号还是被加1，这是因为接收的包中包含SYN或FIN标志位（并不会对有效数据的计数产生影响，因为含有SYN或FIN标志位的包并不携带有效数据） **包3**：客户端对服务端的连接请求进行确认，ACK标志为1，序号为包2的确认号，即：951057940；而确认号为包2的序号加1，即：290218379 + 1 = 290218380 **包4**：这是TCP流中第一个携带有效数据的包（确切的说，是客户端发送的HTTP请求），序列号与前一个ACK报文一致，因为到上个包为止，还没有发送任何数据；确认号也与前一个ACK报文一致，因为客户端没有从服务端接收到任何数据。 包内携带发送给服务器的数据为479个字节。 **包5**：当上层处理HTTP请求时，服务端发送该包来确认客户端在包4中发来的数据，需要注意的是，确认号的值增加了479（479是包4中有效数据长度），即：951057940 + 479 = 951058419。服务端以此来告知客户端端，目前为止，我总共收到了479字节的数据，服务端的序列号保持不变。 **包6**：这个包标志着服务端返回HTTP响应的开始。因为服务端在该包之前返回的包中都不带有有效数据，所以序列号依然不变。该包带有1380字节的有效数据。 **包7**：由于上个数据包的发送成功，TCP 客 户 端 的 序 列 号 增 长 至951058419，从服务端接收了1380字节的数据 ， 客户端的确认号290218380 + 1380 = 290219760。

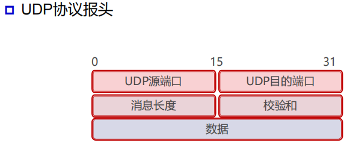


**TCP协议的特点** 全双工连接(full-duplex connection) 该连接的两端有两条彼此独立、方向相反的传输通道 面向连接(connection-oriented) 通信双方在开始传输数据前，必须通过“三次握手”的方式在双方之间建立一条逻辑上的链路（虚电路），用于传输数据 可靠性(reliable) 自动分片；保证传送给应用层的数据顺序是正确的 ；自动过滤重复的封包 ；确认-重传确保数据包可靠到达 面向字节流(byte-stream) 将应用程序和网络传输相分割，为流传输服务提供了一个一致的接口 **拒绝服务（Denial of Service）** DoS攻击是指利用网络协议漏洞或其他系统以及应用软件的漏洞耗尽被攻击目标资源，使得被攻击的计算机或网络无法正常提供服务，直至系统停止响应甚至崩溃的攻击方式。即攻击者通过某种手段，导致目标机器或网络停止向合法用户提供正常的服务或资源访问。利用TCP面向连接的特点:三次握手过程需要存储连接状态，因此会产生系统开销。

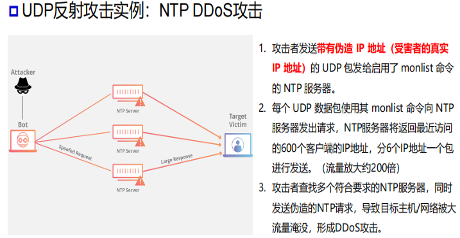


**SYN Flooding** **特点**： 针对TCP/IP协议的设计缺陷进行攻击；发动攻击时，只要很少的数据流量就可以产生显著的效果；攻击来源无法定位；在服务端无法区分TCP连接请求是否合法。 **防御措施**（RFC4987）： 在防火墙上过滤来自同一主机的后续连接；采用SYN Cookie（无法防范全连接拒绝服务） **其它针对TCP协议的攻击** **ACK Flooding：**主机接收到带有ACK状态的数据包，需要检测数据包所包含的连接四元组是否存在，如存在需要检查数据包状态数据是否合法。（消耗资源） **序列号猜测攻击**（会话劫持）：攻击者通过猜测序列号，在TCP会话中插入自己构造的数据包。 **Land攻击：**构造一个SYN包，其源地址和目标地址都被设置成某一个服务器地址；导致接收服务器向它自己的地址发送SYN-ACK消息，结果这个地址又发回ACK消息并创建一个空连接；每一个这样的连接都将保留直到超时。

**UDP协议安全威胁**

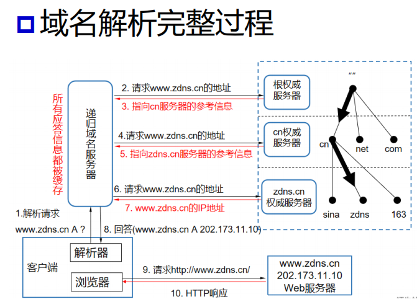


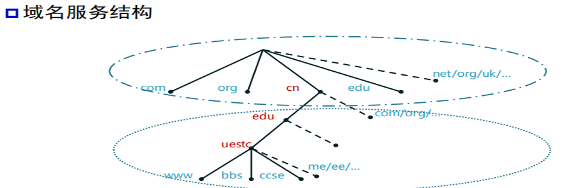


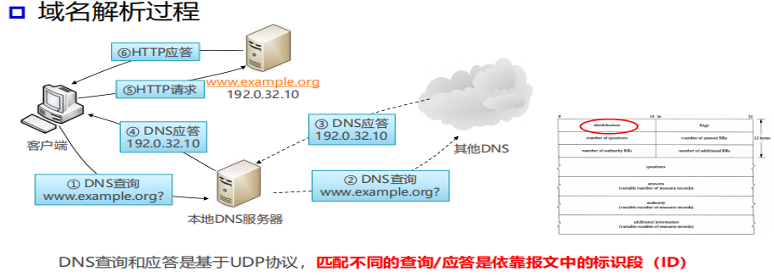




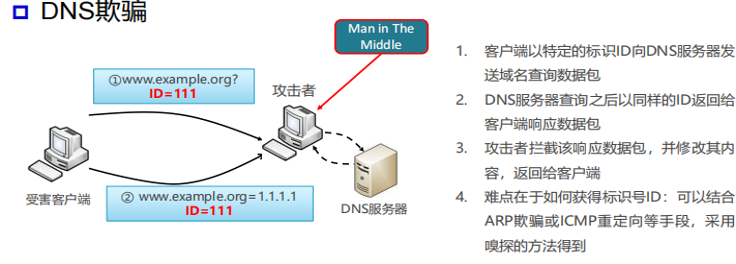
2.5 **应用层安全协议分析** **DNS协议安全威胁**

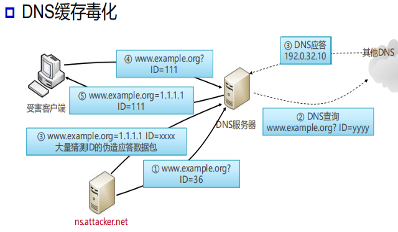


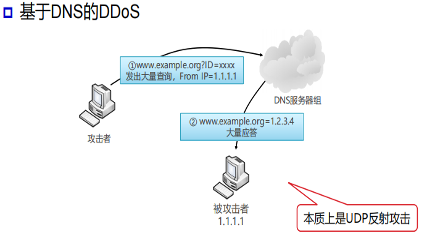




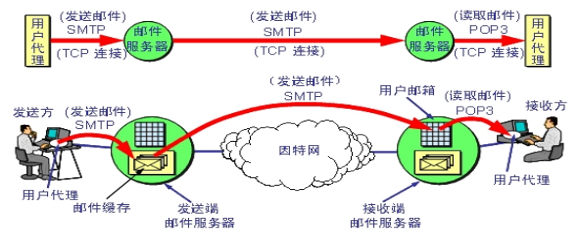
1. 客户端在打开浏览器，输入域名www.zdns.cn。浏览器会发起一个DNS解析请求。
2. 如果本地DNS服务器中找不到结果，则首先向根服务器查询，根服务器里面记录了各个顶级域服务器的IP地址。
3. 当向根服务器请求www.zdns.cn的时，根服务器就会返回cn域名的解析结果。
4. 递归域名服务器向cn权威服务器发起www.zdns.cn的解析请求。
5. cn权威服务器查找并返回zdns.cn的解析结果。
6. 递归域名服务器向zdns.cn的权威服务器发起域名解析请求。7. zdns.cn权威服务器返回www.zdns.cn的解析结果。
7. 递归域名服务器缓存查询结果并向客户端返回查询结果202.173.11.10。9. 客户端访问web服务器获取数据







**电子邮件协议安全威胁**



**电子邮件协议** SMTP：主要负责底层的邮件系统如何将邮件从一台机器传至另外一台机器 POP：目前的版本为 POP3，POP3是把邮件从电子邮箱中传输到本地计算机的协议 IMAP：目前的版本为 IMAP4，是POP3的一种替代协议，提供了邮件检索和邮件处理的新功能 S/MIME:支持邮件加密的传输协议 电子邮件协议的安全：传输安全、发送者身份确认、接收者已收到确认、邮箱炸弹攻击 **HTTP协议安全威胁** **HTTP钓鱼攻击** **跨站攻击**：浏览器对网页的展现是通过解析HTML代码实现的，如果传入的参数含有代码浏览器会解析它而不是原封不动的展示。跨站攻击类型： 持久性跨站（persistent XSS or stored XSS）– 攻击数据存放于服务器。当用户访问正常网页时，服务端会将恶意的指令夹杂在正常网页中传回给用户。 非持久性跨站（non-persistent XSS or reflected XSS）- 当服务端未能正确地过滤客户端发出的数据，并根据用户提交的恶意数据生成页面时，就有可能生成非持久性跨站攻击。 DOM 跨站（DOM-based XSS）– 如果客户端脚本（例如JavaScript）动态生成 HTML的时候，没有严格检查和过滤参数，则可以导致 DOM 跨站攻击。

**如果主机A跳过与主机B建立TCP连接的前两个步骤，直接发送三次握手中最后一个带ACK标志的包，主机B会如何处理？** 它将无法正确地识别这个连接请求，因此会忽略这个单独的带有ACK标志的包

**如果应用程序在释放连接的过程中，由于应用程序异常终止来不及通知TCP协议释放连接，试问在实际情况中应该如何处理这种异常？** 通常，在实际情况中，操作系统会针对这样的情况采取一些机制来处理：TCP超时和重传： TCP协议具有超时和重传机制，如果一端长时间没有收到另一端的响应，它会尝试重新发送数据或发送连接释放的请求。这有助于避免永久性的连接保持状态。操作系统资源管理： 操作系统会周期性地清理无效的或闲置的连接资源，以确保系统资源的有效利用。

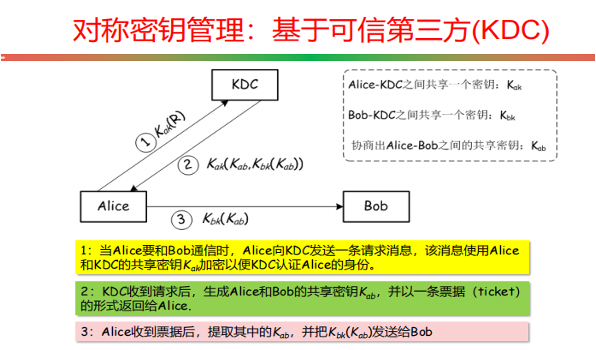
**UDP协议安全威胁产生的根本原因是什么？请举例分析。** UDP的设计目标是简单快速，但也因此存在一些安全威胁，其根本原因主要包括： 缺乏连接状态和可靠性： UDP不像TCP那样有连接状态管理，因此缺乏对数据传输的控制和监视。攻击者可以轻易伪造数据包或进行数据包的欺骗性操作，因为UDP不验证数据包的来源或内容的完整性。 易于伪造IP地址： UDP协议本身不提供对IP地址伪造的保护机制。攻击者可以轻松伪造源IP地址发送UDP数据包，这种做法称为IP欺骗（IP Spoofing）。例如，DNS放大攻击（DNS Amplification Attack）利用UDP的特性，伪造IP地址发送大量的DNS请求，向目标服务器发送大量响应，导致拒绝服务（DDoS）攻击。 易于进行UDP泛洪攻击： UDP泛洪攻击（UDP Flood Attack）利用UDP协议的特性，向目标服务器发送大量的UDP数据包，消耗服务器的网络带宽和处理能力，导致服务不可用。

**域名解析协议中主要存在哪些安全威胁？简要说明威胁过程和原理。** DNS劫持： 攻击者篡改DNS响应，将用户的域名解析请求重定向到恶意网站。这种攻击可能通过在本地网络或路由器上植入恶意DNS服务器实现，或者在用户计算机中安装恶意软件进行DNS劫持。当用户输入一个域名时，他们将被重定向到攻击者控制的恶意网站，可能导致信息泄露或钓鱼攻击。

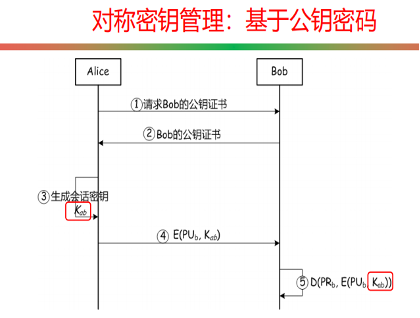
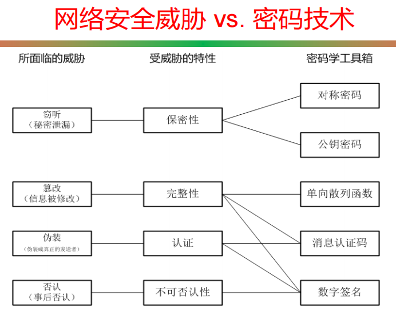
**域名解析协议中主要存在哪些安全威胁？简要说明威胁过程和原理。** DNS投毒： 攻击者在DNS缓存中插入虚假的映射信息，当其他用户向同一DNS服务器请求相同的域名时，他们会获得被篡改的结果。这种攻击可以利用DNS响应的缓存机制，将虚假信息注入到缓存中，影响其他用户的域名解析结果。 DNS DDos： 攻击者尝试大量的DNS查询请求，以耗尽服务器资源或者使得服务器无法响应合法的查询。对DNS服务器发动大规模的查询请求，导致其超载并拒绝正常用户的服务请求。 DNS欺骗： 攻击者在传输过程中伪造DNS响应，使得客户端接收到的IP地址是攻击者所控制的恶意服务器IP地址，而不是正确的IP地址。这可能导致用户被重定向到恶意网站，进行钓鱼攻击或者窃取用户的敏感信息。

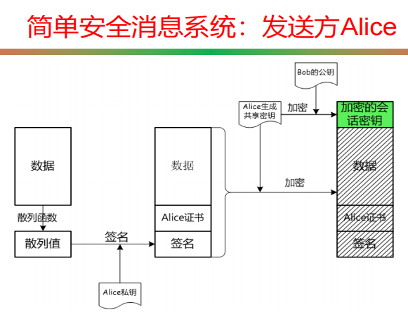
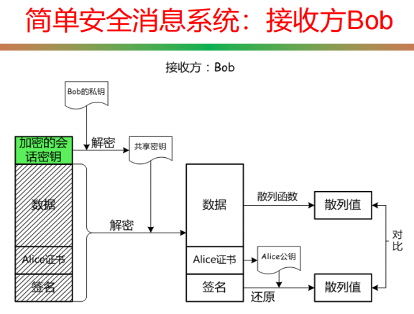
3. **网络安全协议基础** 3.1 **密码学工具箱** 对称加密算法、公钥加密算法、哈希算法、消息认证码、随机数、密钥协商 **对称加密算法：**加密密钥与解密密钥相同。 加解密效率高、适合加密大量数据（AES、3DES、RC4） **公钥加密算法**：加密密钥和解密密钥不同。加解密效率低，不适合加密大量数据，但可用于密钥分发、数字签名等。（RSA、ElGamal、ECC） **哈希（hash）算法：** 接收任意长度的数据，输出定长的散列值（MD5、SHA-1、SHA-2、RIOE-MD160、SHA-3） 密码学哈希函数应具备：映射分布均匀性和差分分布均匀性；单向性；抗冲突性，包括：抗弱冲突性：给定M，计算上无法找到M’，满足H(M) = H(M’)、抗强冲突性：计算上也难以寻找一对任意的M和M’，使其满足H(M) = H(M’) 用途：检测软件是否被篡改、基于口令的加密、数字签名、伪随机数生成器、消息认证码（MAC） **消息认证码（Message Authentication Code）：**确认消息完整性、消息源发认证 输入：任意长度的消息、发送者和接收者之间共享的密钥 输出：固定长度的数据：MAC 总之，消息认证码是一种与密钥相关联的单向散列函数，消息认证码的实现还有多种方式，如GMAC等。 

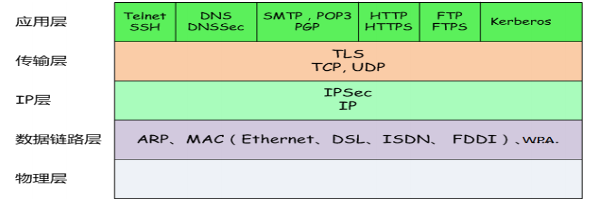
**随机数**： 具有下述属性： 随机性：不存在统计学偏差，是完全杂乱的数列； 不可预测性：不能从过去的数列推测出下一个出现的数； 不可重现性：除非将数列本身保存下来，否则不能重现相同的数列。 用途： 生成密钥：对称密码、消息认证码； 生成初始化向量：分组密码的CBC、CFB和OFB模式； 生成密钥对：公钥密码、数字签名； 生成盐：用于基于口令的密码（PBE） 生成nonce（number once）：用于防御重放攻击及分组密码的CTR模式



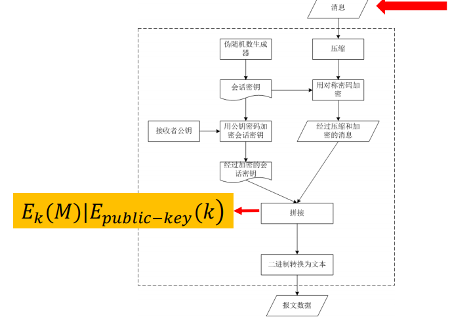
**对称密码的密钥管理** ：基于可信第三方（KDC）、基于密钥协商算法、基于公钥密码 **对称密钥管理：基于密钥协商算法** 密钥协商的思想：通信双方交换生成密钥的素材，并各自利用这些素材在本地生成共享密钥。即使攻击者获取这些素材，也无法生成共享密钥。 密钥协商算法：DH(Diffie-Hellman)、ECDH 公钥密码密钥管理 通过网络直接发送公钥--中间人攻击 数字证书： 公钥； CA数字签名：保证了真实性和完整性；版本；序列号； 签名算法

  3.2 **简单的安全消息系统** Alice拥有一个较大的数据文件，Alice想和Bob通过网络安全地共享这个数据文件。如何设计一个安全消息系统，能够用于发送者Alice和接收者Bob之间进行安全的数据通信？ 采用公钥密码+对称密码：公钥密码进行密钥分发对称密码进行数据加密,不采用公钥密码，效率低不适合加密较大数据文件

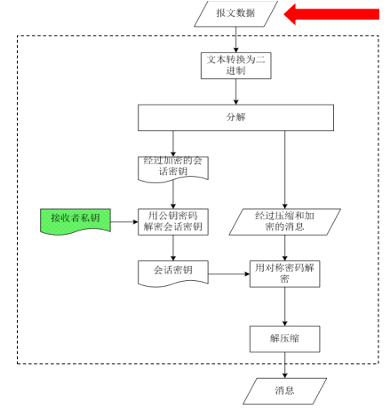
  忽略了身份认证过程：Alice发送数据之前，已经认为接收方就是预期的Bob，但是并没有从技术上来保证。 忽略了算法协商：通信双方使用的加解密算法、散列算法、签名算法必须一致，否则无法正确通信，但是通信双方事前无此协商过程。 忽略了效率问题：能否对数据进行压缩，降低传输的数据量。 **实际网络安全协议的设计思路**：实际的网络安全协议通常把整个协议交互过程划分为协商过程和数据通信两个步骤，并针对两个步骤分别规定了相应的语法、语义和时序。 协商过程包括：**身份认证**；**算法协商**，包括：加解密算法、散列算法、数字签名算法等；**会话密钥协商**，通信双方就保护数据保密性的加密密钥达成共识。 3.3 **网络安全协议概览**



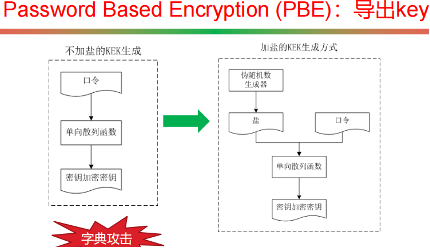
**应用层：** **PGP**： Pretty Good Privacy （PGP）； Philip Zimmermann设计开发；保护邮件通信的隐私性,提供安全服务:保密性、数字签名 **传输层：** **SSL/TLS**：由网景公司发起（Secure Socket Layer，SSL），用于解决浏览器和Web服务器之间通信的安全性IETF进行标准化（Transportation Layer Secure，TLS） 提供安全服务：身份认证、保密性、完整性 **IP层：** **IPSec**：IPSec是IETF开发的一套认证、加密协议，用于解决IP-based网络所缺少的安全性。 RFC2401和2411等，对IPSec进行了具体描述；IPSec是IPv6协议的内置功能，是IPv4的补充功能 IPSec提供以下安全服务：**完整性**：确保接收到的流量没有被篡改过；**保密性**：确保所传输的流量没有被非授权访问；**认证**：特别是源认证，当目的主机接收到一个带特定地址的数据包，能够确保该数据包就是拥有这个特定源IP地址的主机生成；**重放保护**：确保通信双方交互的每个数据包都是不同的。 IPSec 主要通过两个协议来提供上述安全服务Authentication Header (AH) protocol、Encapsulation Security Payload (ESP) protocol （**AH协议**提供源发认证和数据完整性，但是不提供保密性；**ESP协议**提供认证、数据完整性和保密性） **数据链路层：** **WPA/WPA2：**身份认证、保密性 3.4 **OpenPGP** **PGP概述** 安全功能——功能模块 保密性：对称加密 完整性：对消息摘要进行数字签名 身份认证:数字签名 加/解密: 数字签名 压缩：PGP支持数据的压缩和解压，目的是提高数据存储和传输的效率，支持ZIP、ZLIB、BZIP2等格式 电子邮件兼容性：电子邮件系统通常支持ASCII文本格式，而加解密通常是对二进制进行操作，因此需要进行兼容性处理，二进制与Radix-64互相转换 Radix-64编码:基于base64，增加了检测数据错误的校验和。Base64编码是一种可以将任何二进制数据都用A~Z、a~z、0~9、+、/共64个字符外加=（用于末尾填充）来表示的编码方法 **PGP加密和解密**

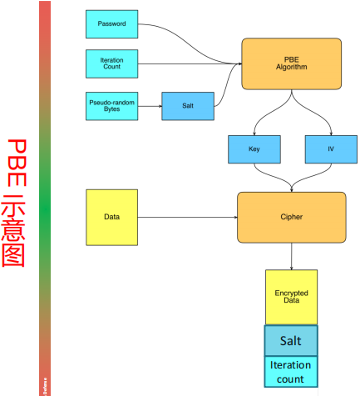
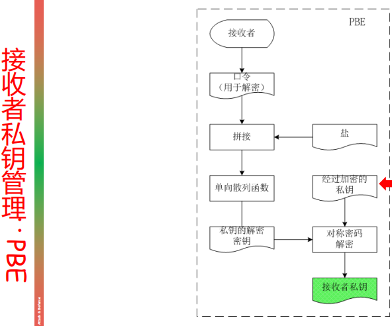
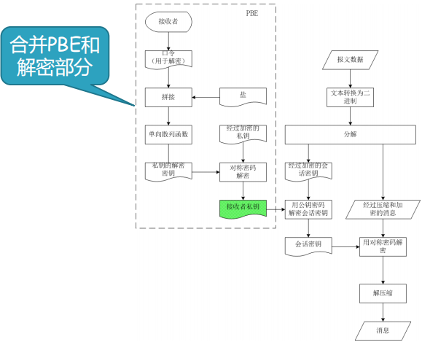
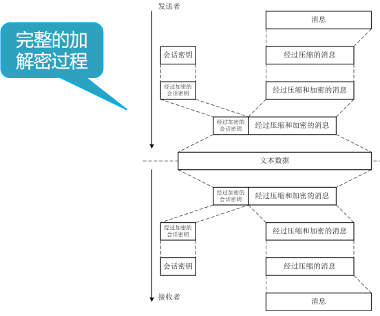


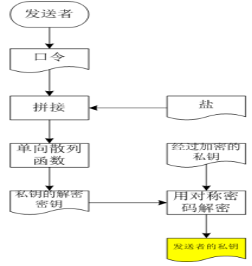
**PGP：加密** 生成和加密会话密钥 用伪随机数生成器生成会话密钥 采用公钥加密算法，用接收者的公钥加密会话密钥压缩和加密消息 压缩消息 使用对称密码对压缩后的消息进行加密，密钥为步骤1中的会话密钥 将加密的会话密钥（步骤2中）与加密的消息（步骤4中）拼接起来 将步骤5中结果转换为文本数据，得到报文数据。



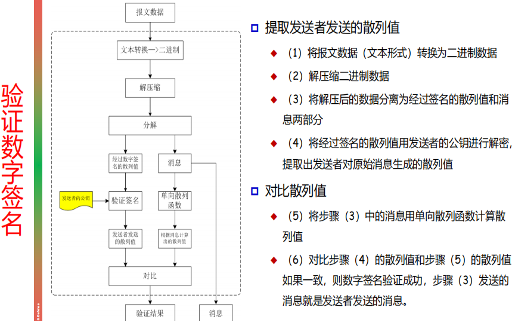
**PGP：解密** 解密私钥 接收者输入解密的口令 通过口令导出解密私钥的密钥 对钥匙串中经过加密的私钥进行解密解密会话密钥 将报文数据（文本形式）转换为二进制数据 将二进制数据分解为两部分：会话密钥的密文和消息的密文 用步骤3中生成的接收者的私钥解密会话密钥解密和解压消息 使用步骤6中生成会话密钥解密消息密文，得到压缩过的消息 对步骤7中的输出进行解压缩 9 得到原始消息 **接收者私钥管理** 私钥——记在脑袋里 记不住：长串随机数 私钥——明文存放计算机上 不安全 加密存放在计算机上 合理：加密私钥的密钥怎么管理？



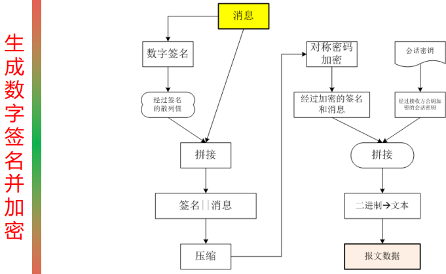
PBKDF2 的 迭 代 次 数 ， 缺省是10000次。PBKDF2生成48字节 其中0~31是key, 32~47是IV     只采用加解密功能，具有消息的保密性，也确保了接收者是正确的，但是无法认证发送者的身份。这也提供了发送匿名邮件的可能性。因此，在只需要消息的保密性的前提下，只用PGP的加解密功能是可以的 **生成和验证数字签名**

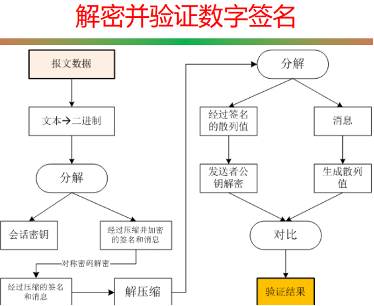


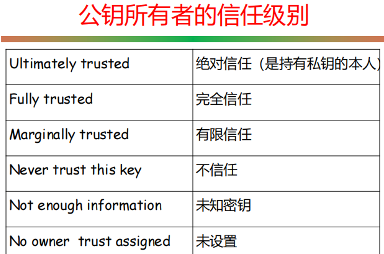
**生成数字签名** 解密私钥 发送者输入用于解密私钥的口令; hash（口令盐）—>解密私钥的密钥(KEK); 解密钥匙串中经过加密的私钥; 用单向散列函数计算消息的散列值; 用步骤3中私钥对上述散列值进行签名; 将步骤5中生成的数字签名与消息进行拼接; 将步骤6中的结果进行压缩; 将步骤7中的结果转换为文本数据，即：报文数据;

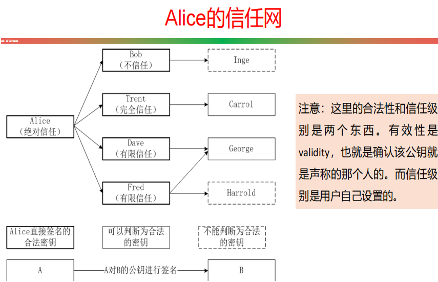
**PGP生成数字签名并加密以及解密并验证数字签名**





**PGP的公钥管理** PGP的正确运行依赖于公钥的有效性。 PGP没有假定信任任何机构，即使是国家 不依赖于PKI PGP采用信任网（web of trust）来确保公钥的有效性 信任网：基本原则,通过自己的数字签名进行确认、通过自己完全信任的人的数字签名进行确认、通过自己有限信任的多个人的数字签名进行确认 **通过自己的数字签名进行确认** Alice和Bob在一次meeting上认识，分别时，Bob交给Alice一个U盘，并说“以后email联系，这是我的公钥”。Alice回家后，将Bob的公钥从U盘中导入到自己的PGP公钥串中。Alice确信U盘中的公钥就是Bob本人的，因此，Alice对这个公钥加上自己的数字签名，以证明这个公钥是有效的。随后，Alice收到来自Bob的email，这封邮件带有Bob的数字签 名，为了验证Bob的数字签名，PGP需要执行以下步骤。 PGP从Alice的公钥串中寻找Bob的公钥。（根据数字签名的原理，验证某个数字签名，需要对应的公钥） Alice的公钥串中有Bob的公钥。（因为，Alice在之前导入过Bob的公钥） （但是，这个公钥合法吗？）PGP发现公钥串中Bob的公钥带有Alice的数字签名。（Alice在导入Bob的公钥的时候，对它做了数字签名） 为了验证Alice的数字签名，PGP需要从Alice的公钥串中寻找Alice自己的公钥。（Alice的公钥串中也包含了Alice自己的公钥） PGP使用Alice自己的公钥对Bob的公钥上的数字签名进行验证。如果成功，则表明这的确是Bob的公钥。 PGP使用合法的Bob的公钥对邮件上附带的Bob的数字签名进行验证。如果验证通过，则表明该email确实来自于前段时间开会时遇到的Bob所发。 **通过自己完全信任的人的数字签名进行确认** Alice有个男朋友叫Trent。在Alice的公钥串中，包含带有Alice签名的Trent的公钥。Alice完全信任Trent，可以认为“Trent介绍的人也是可信的”。因此，Alice设置Trent的公钥为“我完全信任Trent的数字签名”这一状态，并加上自己的数字签名。 也就是一旦验证是Trent签名的公钥，就是有效的公钥。在PGP中，用户可以设置对每个公钥所有者的“所有者信任级 别”（owner trust），因为Alice完全信任Trent，因此把Trent的“所有者信任级别”设置为“完全信任（Fully Trusted）” 假设某一天，Alice收到了自称来自Carrol的邮件，邮件中附带有Carrol的公钥，而且这个公钥带有Trent的数字签名。Alice把Carrol的公钥导入自己的公钥串时，Alice的PGP通过Trent的公钥验证了Carrol的公钥的有效性。因为Alice完全信任Trent，因此Alice的PGP会认为Carrol的公钥是有效的。 在这一场景中，Alice信任Trent，Trent通过给Carrol的公钥数字签名而起到了“介绍人”的角色，从而Alice可以认为Carrol的公钥是有效的。注意：因为Alice的公钥串中没有Carrol的公钥，因此无法通过Alice的公钥串来直接确认Carrol的邮件。 

**通过自己有限信任的多个人的数字签名进行确认** 假设Alice有两个男朋友，分别叫Dave和Fred。在Alice的公钥串中，包含有带Alice签名的上述两人的公钥，而且Alice把Dave和Fred的所有者信任级别都设置为“有限信任”。某一天，Alice收到了来自George的公钥，该公钥带有Dave和Fred的数字签名，Alice的PGP会确认该公钥是有效的，确实属于George。如果仅有Dave或者Fred之一对该公钥进行数字签名，因为Alice对Dave和Fred只是有限信任，因此不能确认该公钥是有效的。只有当2个或以上的有限信任的人对某个公钥签名时，才能确认该公钥是有效的



**PGP总结**

