密码应用安全的技术体系探讨

林璟锵　　荆继武
(中国科学院数据与通信保护研究教育中心　北京　100093)
(信息安全国家重点实验室(中国科学院信息工程研究所)　北京　100093)
(中国科学院大学网络空间安全学院　北京　100049)
(linjingqiang@iie.ac.cn)

The Taxonomy Towards the Security Application of Cryptography

Lin Jingqiang and Jing Jiwu
(*Data Assurance and Communications Security Research Center*, *Chinese Academy of Sciences*, *Beijing* 100093)
(*State Key Laboratory of Information Security* (*Institute of Information Engineering*, *Chinese Academy of Sciences*), *Beijing* 100093)
(*School of Cyber Security*, *University of Chinese Academy of Sciences*, *Beijing* 100049)

Abstract　Cryptography plays an important fundamental role in cyber security. Applying cryptography in computer and network systems to implement security services has improved the security of cyber space. The application of cryptography in cyber space, requires the consideration of the view of cryptography from the point of view of computer and network security, to establish the relationship between rigorous but abstract cryptography and complex but concrete information systems. This paper discusses the taxonomy of the secure application of cryptography, by analyzing the influences among data, systems, and entities. We attempt to answer the question: when cryptography theory is ready, which technical issues shall be solved towards the secure application of cryptography in computer and network systems? We list the following issues: 1) choose suitable cryptographic algorithms, work modes and cryptographic protocols, 2) maintain reasonable cryptographic keys, 3) generate secure random numbers, 4) implement and deploy cryptographic protocols correctly, 5) bind cryptographic keys to entities, 6) ensure the security of cryptographic keys, and 7) enforce the use control of cryptographic computations. Based on the related works, we describe each of these technical issues detailedly.

Key words　applied cryptography; network security; system security; cyber security; cryptographic key security

摘　要　密码学是网络空间安全技术的重要组成,发挥了基础性的核心作用.在计算机和网络系统中应用密码学原理,设计和实现安全服务,极大地提高了网络空间的安全性.在网络空间中应用密码算法和密码协议,需要从计算机和网络系统的角度来考虑密码技术,在严谨而抽象的密码学与复杂

收稿日期:2018-11-15
基金项目:国家自然科学基金项目(61772518);国家重点研发计划网络空间安全重点专项(2017YFB0802100)

而具体的信息系统之间建立联系. 从数据、系统、实体等之间关系的角度出发，初次尝试探讨密码应用安全的技术体系；即在密码学理论已经完备的前提下，在计算机和网络系统中应用密码学原理，应该重点解决哪些方面的技术问题，列出了密码应用安全研究中需要完成的工作：1）选择合适的密码算法、工作模式和密码协议；2）维护合理的密钥参数；3）产生安全的随机数；4）以正确的方式实现和使用密码协议；5）绑定密钥与实体；6）确保密钥安全；7）实施密码计算的使用控制. 结合已有的公开研究成果，详细论述了每一方面研究的问题和内容.

关键词 应用密码学；网络安全；系统安全；网络空间安全；密钥安全

中图法分类号 TP309

1 密码应用安全

在计算机和网络系统中应用密码学原理，利用密码技术实现计算机和网络系统的安全，是网络空间安全的重要工作之一. 然而，各种安全事件和研究成果表明，大量基于密码学原理的安全服务仍存在漏洞或缺陷，并不能达到预期的、密码学理论强度的安全保障. 现实情况表明，为了在计算机和网络系统中更紧密地融合密码技术，使得密码技术更充分地在网络空间中发挥安全作用，仍然还有大量的技术研究工作需要我们来完成；这一部分技术研究，本文称之为“密码应用安全”.

在具体实践中，密码应用安全研究需要从实际运行的计算机和网络系统出发，融合密码算法和密码协议的理论成果，在技术上连接二者. 所以，从知识体系和研究风格而言，密码学（尤其是理论密码学）更接近于理科，系统安全和网络安全更接近于工科，密码应用安全则是介于理科和工科之间的基础技术研究.

2 密码应用安全的技术体系探讨

在本文中，我们不考虑特定的应用服务，仅仅考虑通用的密码学应用场景，从数据、系统、实体 3 个角度来分析密码技术应用中的安全问题，尝试得到密码应用安全的技术体系，即在各种密码学理论研究成果已经就绪的前提下，网络空间还需要解决哪些安全技术问题才能够保证密码技术成功应用，以实现预期的安全目标；在计算机和网络系统中应用密码学原理，我们应该考虑哪些方面的技术问题.

如图 1 所示，密码算法、工作模式和密码协议是典型的传统密码学理论研究方向：

1）对于基础性的核心密码算法（例如，对称加密算法、公钥加密算法、数字签名算法等），最核

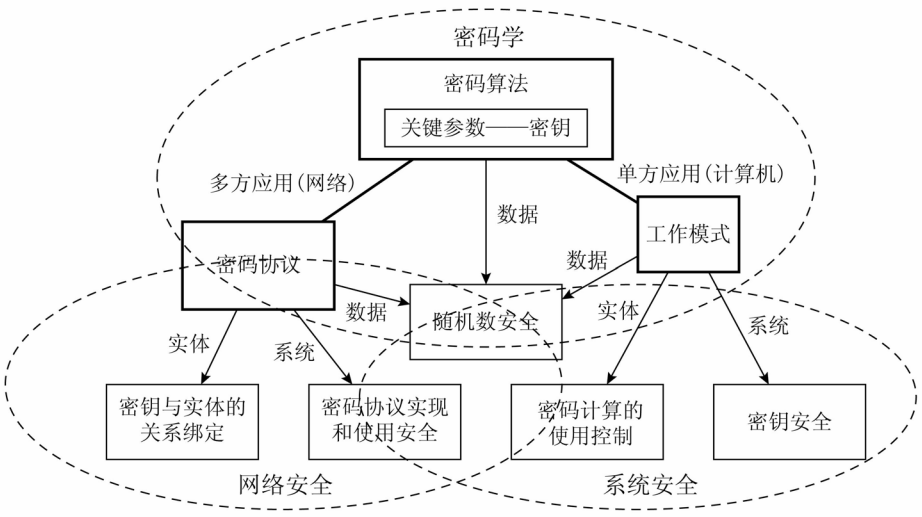

图 1 密码应用安全技术体系

心、最关键的参数是密钥. 按照 Kerckhoffs 原则，密码体制的安全性应仅依赖于密钥的机密性，密码算法的其他参数都是公开的.

2) 在使用密码算法时，即使仅仅是单方利用密码算法进行数据处理时也需要选用工作模式(例如，对称加密算法的工作模式、公钥加密算法和数字签名算法的数据填充等).

3) 在利用密码算法构建多方之间的安全通信和安全服务时，涉及各种密码安全协议的设计和分析. 例如，各种身份鉴别协议、TLS 协议等都是常用的密码安全协议. 针对不同应用场景(例如，云计算、区块链、物联网、大数据等)，设计专门的密码安全协议也是应用密码学研究的重要内容之一.

我们进一步从数据、系统、实体 3 个角度来分析在密码学理论成果在应用中需要考虑的安全问题.

1) 对于密码算法，从数据角度而言，最重要的密钥数据应源自随机数，保证攻击者不可预测.

2) 各种密码算法工作模式和密码协议通常也需要安全的随机数作为执行过程中的参数.

3) 在密码协议的设计和分析中，通常使用不同的密钥来区分不同的实体，密钥与实体之间的关系绑定直接作为基本假设. 例如，假设 Alice 正确获得 Bob 的公钥；即 Bob 与其公钥的绑定关系是明确的、公开已知的.

4) 在密码协议过程中，要求各网络实体执行相应的操作，体现为计算机系统上的多个进程或线程任务. 在不同的计算平台上，各任务的处理过程和输入输出、任务之间的数据传递等，并不一定必然满足密码协议对运行环境的隐含条件和假设要求.

5) 对于在计算机系统上执行计算的密码算法(按照特定工作模式或者密码协议的要求)，从计算机系统安全的角度而言，要求确保攻击者不能读取访问密钥，以满足密码算法的最基本要求.

6) 计算机系统上执行的密码计算应直接或间接地体现实体的操作意志，得到用户的授权，以实现高安全强度的密码计算使用控制.

综合以上分析，为了在计算机和网络系统中发挥密码技术的安全作用，密码应用安全技术研究需要考虑如下.

2.1 选择合适的密码算法、工作模式和密码协议

研究安全的密码算法、工作模式和密码协议是密码学理论研究的重要内容. 对于大多数密码应用安全来说，应该选择公开的、经过全面深入分析的、标准化的密码算法、工作模式和密码协议. 而且，信息系统应该具备密码升级能力，尤其对于长期使用的系统，一旦某些密码算法、工作模式和密码协议出现问题需及时更换.

在实际运行的计算机和网络系统彻底实施以上要求并不容易，一方面要及时理解密码学研究的理论成果，还需要在系统中留有足够的安全余量，保持算法升级能力. 例如，近年来的 Hash 算法攻击分析[1-2]就使得原有 MD5 算法和 SHA-1 算法从安全变为不安全；2011 年的互联网统计分析表明[3]，大量网站仍然使用由不安全密码算法签发的数字证书(MD5 和 RSA-1024 算法)；2016 年的统计数据表明[4]，不安全的 RC4 对称密码算法在 SSL/TLS 通信中仍然大量使用.

密码算法的工作模式研究在密码学理论研究中一直是重要的方向. 不论是对称密码算法、非对称密码算法或者 Hash 算法，在处理数据时都需要考虑工作模式/填充模式. 相比于密码算法，工作模式相关研究成果的应用实施不易理解，难度更大. 在大量公开资料和教科书中都以经典的 AES 算法和 RSA 算法为例来说明对称密码算法和公钥密码算法. 然而，在常见的举例中，AES 算法表现为密钥对 128 b 明密文的处理过程，RSA 算法表现为直接简单的、易于理解的模幂计算，都没有考虑工作模式/填充模式. 2018 年发现的 RSA 算法使用问题[5]和 2014 年提出的 SSL/TLS DROWN 攻击[6]都是针对不安全的 RSA 算法数据填充，文献[7]也讨论了数据存储加密中不同工作模式的相应安全问题.

对于密码协议，以广泛使用的 SSL/TLS 协议为例，也可以看出，不安全的历史版本仍然在实际系统中大量使用[4,8].

2.2 维护合理的密钥参数

按照信息系统的安全需求，选择合适的密钥长度；同时，考虑系统运行期间的密钥更新(周期性执行、按需执行以及安全事件时的密钥更新).

不同密钥长度的相同算法代表了不同的安全强度. RSA 算法的安全密钥长度一直随着技术研

究进展而不断调整[8-13]. 文献[8]通过对 Diffie-Hellman 密钥交换协议的技术分析，推测 NSA 有能力解密大量 SSL/TLS 加密流量，也会改变 SSL/TLS 协议的密钥参数.

为了前向/后向安全及考虑到可能发生的安全事件，密钥需要定期更换，所以密码应用安全系统需要支持自动或者手动方式的密钥更新功能. NIST SP 800-57 给出了各种应用场景推荐使用的密钥安全强度和密钥更新周期[13]，包括数据传输、数据存储、数字签名、鉴别、授权、密钥传输和封装、密钥协商、随机数等不同应用场景.

2.3 产生安全的随机数

随机数在密码算法、工作模式和密码协议中大量使用. 必须以安全的方法、安全的流程来产生随机数，才能将其用作密钥或者其他参数.

在信息系统中，高速地产生安全的随机数面临种种挑战. 1996 年，研究人员就发现，Netscape Browser 的 SSL 协议实现使用了有问题的随机数生成方法[14]，攻击者可以预测密钥. 在 20 年前，文献[15]总结了若干实际系统中的随机数问题，并给出了在计算机系统中产生安全随机数的技术建议. 文献[16]分析了 Windows 操作系统随机数产生函数 CryptGenRandom 的实现.

目前研究成果中有多种方法来产生随机数，而且其中有一些已经成为标准[17-18]. 标准化的方法也有可能存在安全问题；例如，著名的 Dual EC 随机数产生方法就被广泛猜测设计上有后门[19]：Dual EC 随机数方法发布在 2006 年的 NIST SP 800-90A 标准中，但是在 2014 年的标准新版本中去掉了 Dual EC 随机数方法[17]；ANSI X9.17/X9.31 随机数产生标准，如果在系统实现中使用固定密钥也容易导致随机数猜测攻击[20]. 研究成果表明，现有随机数检测评估标准也存在缺陷[21-22]，不能完全正确地评估随机性.

即使随机数产生方法是安全的，在实现中保证不同的计算机系统之间有足够的差异性仍很困难. 如何保证不同系统使用不同的随机数用作密钥，2012 年的研究表明[23-24]，Internet 上公开使用的数字证书和 PGP/SSH 密钥中有大量的重复密钥参数或者重复使用的随机数.

2.4 以正确的方式实现和使用密码协议

通常而言，在实现密码协议时我们会关注协议所规定的网络实体之间的数据传递和基于密码算法的数据处理，然而，还需要分析实际的系统运行环境与密码协议的初始条件、假设要求之间的符合性. 二者之间的差距就会导致密码协议在实现和使用上的安全问题. 这类的安全问题表现为系统软件漏洞，但是应该更多地从密码应用安全的角度来审视和解决.

在现有的计算机网络系统中广泛使用的密码安全协议主要有 SSL/TLS 协议和单点登录协议. 各种 SSL/TLS 公开源代码已经在大量网络系统中用于通信安全，然而，在很多方面仍然存在问题[4,25-27]：算法配置、证书配置、与应用层的协作、证书/密钥重复使用、操作流程复杂、代理劫持等等. 各种问题使得 SSL/TLS 协议并不能完全达到预定的安全目标(包括身份鉴别、数据机密性或数据完整性).

目前广泛使用的单点登录协议包括 OpenID，OAuth，SAML 等. 在实现和使用中也存在问题，原因包括：不同程序片段对数字签名数据的不同理解[28]、鉴别凭证和授权凭证的错误混用[29-30]、关键数据在浏览器消息处理和传递过程中的泄露和篡改[29-31]等. 以上问题有些来自协议实现和使用中的理解偏差，也由于实际计算机和网络系统在消息处理和传递过程中的复杂性. 例如，在 XML 数字签名数据处理中，数字签名验证和数据解析的分离处理[28]. 对于浏览器传递和处理的消息，有多种正常方法或者攻击手段可以修改和读取[29-31]. 进一步，单点登录和统一认证授权协议实现、从 B/S 模式转到智能移动终端 APP 模式也会引入新的问题[32]，例如，APP 使用 WebView 访问身份服务提供方时的数据泄露、在不可信环境中 APP 的敏感信息泄露等.

2.5 绑定密钥与实体

在密码学研究中通常直接使用密钥来代表不同身份的实体，假定密钥与实体之间的绑定关系是明确的、公开已知的. 由于公钥密码学和 PKI 的发展，在实际运行系统中实体与密钥的绑定关系大量体现为 PKI 数字证书. 正确的 PKI 数字证书验证关系到大量实际运行系统的安全性. 对于其他类型的密码算法(例如，对称密码算法、基于标识的密码算法等)，也尤其应关注其中的初始化密钥分发、初始化参数等步骤.

现有研究发现，大量 SSL/TLS 软件实现并没有正确地验证网络实体与公钥/数字证书的绑定关系[33-34]. 在数字证书验证过程中，如果没有检查根 CA 证书配置和实体身份标识，即使启用了 SSL/TLS 协议也仍然会有遭受中间人攻击的风险. 数字证书验证过程的复杂性（尤其是各种数字证书扩展），也会影响 PKI 证书链的验证正确性[35-37]. 此类型的软件逻辑漏洞不同于常规的软件漏洞，解决方案也不相同.

不少安全事件表明，CA 系统在某些非常极端的复杂攻击情况下会签发含有虚假信息的数字证书. 提高 PKI 数字证书的可信程度、实现网络实体与公钥的可信绑定也是当前的重要技术研究内容，包括：数字证书/公钥 Pinning [38]、证书持有者的控制和确认[39-40]、数字证书签发操作的公开审计[41]、数字证书的多重认证[40,42]、不同网络路径的数字证书对比[43]、证书服务范围限定规则[44-45]等等.

2.6 确保密钥安全

在计算机系统中有多种攻击方法可以越权获得计算任务中的敏感数据. 作为关系到大量应用层数据的、关键的重要敏感数据，我们应该针对密钥实施专门的、比应用层数据更高强度的安全措施. 然而，在各种软件系统中密钥与其他数据都是同等处理、都是程序数据段中的一部分.

在计算机系统中密钥数据以及其他敏感数据面临着各种攻击. 首先，密码计算过程中的各种侧信道攻击[46-54]一直以来都是密码工程的重要研究方向. 其次，作为计算机进程中的内存数据，密钥也面临各类系统攻击和物理攻击，包括 Cold-Boot 攻击[55]、DMA 攻击[56-57]、计算机系统功能导致的数据扩散和软件漏洞导致的内存信息泄露[58-59]（例如著名的 OpenSSL 心脏出血）.

2014 年，Intel 公司推出 SGX 机制[60]，实现了由 CPU 硬件支持的、高强度的隔离计算环境. 在 SGX 执行环境中的数据只在 Cache 中出现，交换到内存芯片时会自动由 CPU 加密，可以抵抗恶意操作系统以及恶意进程读取内存中的敏感数据. Intel SGX 机制利用密码技术在 CPU 中创建硬件支持的安全计算环境，我们也可以在 SGX 执行环境中实现密码算法，由 SGX 机制来保护密钥数据. 但是，近年研究成果表明，SGX 执行环境仍然面临着多种侧信道攻击获取密钥等敏感数据[61-63]、控制流劫持[64-65]等安全威胁. 2018 年，Meltdown 漏洞和 Spectre 漏洞的发现引起了网络空间安全各界的极大关注[66-67]，该漏洞影响了不同厂商、不同型号的大量 CPU，使得攻击者非授权地读取数据（包括密钥等敏感数据），也可以突破 SGX 机制的保护. 这一事件显示，随着 CPU 硬件承载了更多的复杂功能，安全漏洞也会逐渐从软件推进到硬件，密钥安全的攻防研究也会进入新的阶段.

2.7 实施密码计算的使用控制

在计算机系统安全研究中可信用户界面是其中的重要组成. 同样，密码计算使用控制应该保证每次密码计算都是实体操作意志的体现，而且应该达到与密码算法相近的高安全强度.

目前而言，最为常用的密码计算使用控制就是操作口令. 然而，对于没有用户/管理员实时参与操作、或者长期连续调用密码计算的大量应用场景，操作口令的安全性有待提高. 近年发生的 CA 系统签发虚假数字证书的安全事件也表明，即使密钥没有丢失攻击者也可以恶意地调用密码计算.

文献[68-69]早在 2001 年和 2002 就讨论利用 RSA 门限密码算法，结合在线的半可信系统，实现实时的、零延迟的密钥撤销. 类似方案也可以用来支持 SM2 国产密码算法的、实现智能移动终端的密码计算访问控制[70]. En-ACCI 方案[71]利用虚拟机自省技术，检验密码计算调用进程的完整性，实施控制策略，适用于保护虚拟化环境的密码计算服务. 此外，CASTLE 方案提出的完全物理隔离、利用二维码图片交换数据的方式是很有意思的探索，但是只能用在需求量极少的密码计算服务[72].

2.8 讨 论

在以上列出的 7 点研究内容中，解决第 1 点和第 2 点的重点在于如何让应用系统的研发人员及时地获得并理解密码学理论研究成果，并快速完善地在计算机和网络系统中实施. 第 3 点的随机数安全问题也是传统密码学理论研究的重要方向之一. 然而，完成在各种计算机和网络系统中、在各种应用场景中找到行之有效的解决方案，仍然有巨大的技术挑战.

第 4～7 点的技术问题更多表明了密码学和系统安全/网络安全之间的研究空白. 一方面，在密码学理论研究中，密码技术与系统、与实体之间的联系大都直接作为假设要求而存在；另一方面，通

用的计算机和网络系统如果没有专门的设计和实现，难以完全满足密码应用要求的运行环境假设（而且还需要对可执行程序、浏览器、脚本、移动APP等不同运行环境分别处理）. 密码应用安全的措施相比通用的软件安全、操作系统安全和网络安全，有着特殊的技术挑战和解决方案.

我们上文讨论的技术体系没有考虑特定的应用场景，仅仅考虑通用的应用场景. 对于特定的应用场景还会有新的技术挑战，或者某些研究方向和技术挑战会有明显变化. 例如，在比特币系统中直接使用公钥数据来标识不同的实体，密钥与实体的关系绑定问题就明显弱化；在专用密码芯片硬件中只运行密码计算任务，密钥安全问题相对容易解决；对于密码技术的金融支付应用，密码计算的使用控制是非常重要的，反过来，在时间戳服务中，时间戳消息的数字签名基本上就是自动执行，无需人工干预.

3 与现有研究方向的关系

以上我们从密码学理论研究的3个主要方向（密码算法、工作模式、安全协议）出发，分析它们在密码应用中与数据、系统、实体之间的联系，尝试得到密码应用安全的技术体系. 下面，我们简单说明一些常见网络空间安全研究方向与上述技术体系的关系.

如图2所示，传统密钥管理研究的主要内容是通过多方的安全协议实现密钥的管理（注册、生成、更新、撤销、恢复等）；密钥管理也是密码学研究的重要组成.

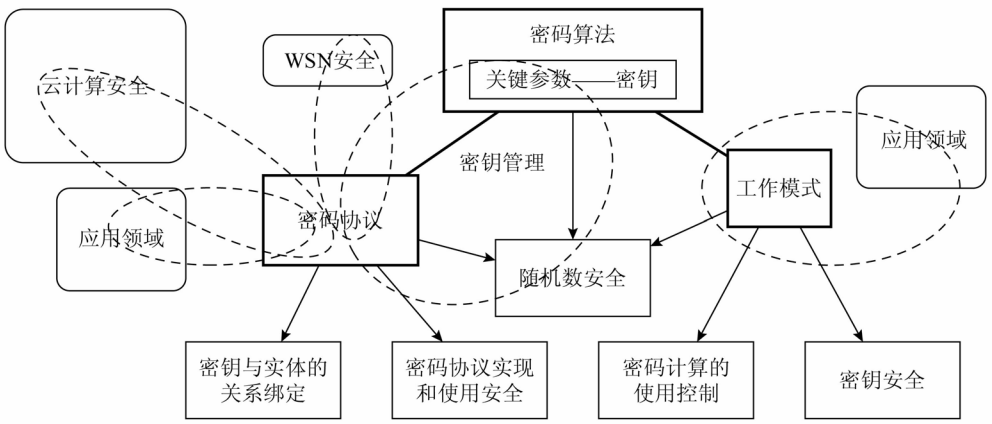

图2 密码应用安全技术体系与现有研究方向

云计算安全的研究内容之一是针对云计算环境设计密码协议，在考虑不完全可信的云服务器的前提下，实现租户数据和计算的安全性涉及云计算安全与密码协议. 无线传感器网络WSN安全的主要研究内容是利用密码协议实现针对WSN环境的密钥管理，以及在此基础上实现的身份管理. 除了云计算和WSN，在其他应用领域也有大量基于密码学原理的安全解决方案，所以在这些领域也有相关的系统安全和网络安全研究成果，利用密码算法、工作模式或密码协议的特性实现相应的安全服务.

4 总 结

在计算机和网络系统中应用密码技术，能有效提高网络空间的安全性. 我们尝试总结密码应用安全的技术体系，列出了7点研究内容；大部分研究内容在公开文献中已有涵盖和涉及，我们尝试将其总结整理为密码应用安全的技术体系.

本文的尝试更多是希望由此带动更全面的密码应用安全技术研究. 由于成稿时间所限，总结的技术体系在内容上会有所遗漏，有所偏差，欢迎大家提出意见，共同探讨.

参 考 文 献

[1] Wang X, Yu H. How to break MD5 and other Hash functions [G] //LNCS 3494: Advances in Cryptology—EUROCRYPT. Berlin: Springer, 2005: 19-35

[2] Wang X, Yin Y L, Yu H. Finding collisions in the full SHA-1 [G] //LNCS 3621: Advances in Cryptology—CRYPTO. Berlin: Springer, 2005: 17-36

[3] Holz R, Braun L, Kammenhuber N, et al. The SSL landscape: A thorough analysis of the x. 509 PKI using active and passive measurements [C] //Proc of the 11th ACM SIGCOMM Internet Measurement Conf. New York: ACM, 2011: 427-444

[4] Holz R, Amann J, Mehani O, et al. TLS in the wild: An Internet-wide analysis of TLS-based protocols for electronic communication [C] //Proc of the 23rd Annual Network and Distributed System Security Symp. Virginia: ISOC, 2016

[5] Knockel J, Ristenpart T, Crandall J. When textbook RSA is used to protect the privacy of hundreds of millions of users [EB/OL]. (2018-02-09) [2018-11-15]. https://arxiv.org/abs/1802.03367

[6] Aviram N, Schinzel S, Somorovsky J, et al. DROWN: Breaking TLS using SSLv2 [C] //Proc of the 25th USENIX Security Symp. Berkeley, CA: USENIX Association, 2016

[7] Ball M V, Guyot C, Hughes J P, et al. The XTS-AES disk encryption algorithm and the security of ciphertext stealing. [J]. Cryptologia, 2012, 36(1): 70-79

[8] Adrian D, Bhargavan K, Durumeric Z, et al. Imperfect forward secrecy: How Diffie-Hellman fails in practice [C] //Proc of the ACM Conf on Computer and Communications Security 2015. New York: ACM, 2015: 5-17

[9] Lenstra A K, Verheul E R. Selecting cryptographic key sizes [J]. Cryptology, 2001, 14(4): 446-465

[10] NESSIE Consortium. Portfolio of recommended cryptographic primitives [EB/OL]. (2003-03-27) [2018-11-15]. http://cgi.di.uoa.gr/~halatsis/Crypto/Bibliografia/Systems&Standards/Nessie/decision-final.pdf

[11] Kaliski B. TWIRL and RSA key size [EB/OL]. (2003-05-06) [2018-11-15]. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.77.4447&rep=rep1&type=pdf

[12] Orman H, Hoffman P. IETF RFC 3766: Determining strengths for public keys used for exchanging symmetric keys [OL]. [2018-11-15]. https://tools.ietf.org/html/rfc3766

[13] Barker E, Barker W, Burr W, et al. Recommendation for key management part 1: General (revision 3) [R]. Gaithersburg: NIST Special Publication, 2012

[14] Goldberg I, Wagner D. Randomness and the netscape browser [J]. Dr Dobb's Journal - Software Tools for the Professional Programmer, 1996, 21(1): 66-71

[15] Gutmann P. Software generation of practically strong random numbers [C] //Proc of the 7th USENIX Security Symp. Berkeley, CA: USENIX Association, 1998: 243-257

[16] Dorrendorf L, Gutterman Z, Pinkas B. Cryptanalysis of the windows random number generator [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2007: 476-485

[17] Barker E B, Kelsey J M. Recommendation for random number generation using deterministic random bit generators (revised) [EB/OL]. (2015-06-24) [2018-11-15]. https://www.nist.gov/publications/recommendation-random-number-generation-using-deterministic-random-bit-generators-2

[18] Rukhin A, Soto J, Nechvatal J, et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications [OL]. [2018-11-15]. https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpub lication800-22r1a.pdf

[19] Checkoway S, Fredrikson M, Niederhagen R, et al. On the practical exploitability of dual EC DRBG in TLS implementations [C] //Proc of the 23rd USENIX Security Symp. Berkeley, CA: USENIX Association, 2014: 319-335

[20] Kelsey J, Schneier B, Wagner D, et al. Cryptanalytic attacks on pseudorandom number generators [C] //Proc of the International Workshop on Fast Software Encryption. Berlin: Springer, 1998: 168-188

[21] Zhu S, Ma Y, Lin J, et al. More powerful and reliable second-level statistical randomness tests for NIST SP 800-22 [C] //Proc of the Int Conf on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2016: 307-329

[22] Zhu S, Ma Y, Chen T, et al. Analysis and improvement of entropy estimators in NIST SP 800-90B for non-IID entropy sources [J]. IACR Trans on Symmetric Cryptology, 2017 (3): 151-168

[23] Lenstra A K, Hughes J P, Augier M, et al. Public Keys [M]. Berlin: Springer, 2012: 626-642

[24] Heninger N, Durumeric Z, Wustrow E, et al. Mining your Ps and Qs: Detection of widespread weak keys in network devices [C] //Proc of the 21st USENIX Security Symp. Berkeley, CA: USENIX Association, 2012

[25] Fahl S, Acar Y, Perl H, et al. Why eve and mallory (also) love webmasters: a study on the root causes of SSL misconfigurations [C] //Proc of the 9th ACM Symp on Information, Computer and Communications Security. New York: ACM, 2014: 507-512

[26] Krombholz K, Mayer W, Schmiedecker M, Weippl E. I have no idea what I'm doing—On the usability of deploying HTTPS [C] //Proc of the USENIX Security Symp. Berkeley, CA: USENIX Association, 2017: 1339-1356

[27] de Carnavalet X C, Mannan M. Killed by proxy: Analyzing client-end TLS interception software [C] //Proc of the Network and Distributed System Security Symp. Virginia: ISOC, 2016

[28] Somorovsky J, Mayer A, Schwenk J, et al. On breaking SAML: Be whoever You want to be [C] //Proc of the 21st USENIX Security Symp. Berkeley, CA: USENIX Association, 2012

[29] Li W, Mitchell C J. Analysing the security of Google's implementation of OpenID connect [C] //Proc of the Int Conf on Detection of Intrusions and Malware, and Vulnerability Assessment. Berlin: Springer, 2016: 357-376

[30] Zhou Y, Evans D. SSOScan: Automated testing of Web applications for single sign—On vulnerabilities [C] //Proc of the 23rd USENIX Security Symp. Berkeley, CA: USENIX Association, 2014

[31] Wang R, Chen S, Wang X. Signing me onto your accounts through Facebook and Google: A traffic-guided security study of commercially deployed single-sign-on Web services [C] //Proc of the IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2012: 365-379

[32] Wang H, Zhang Y, Li J, et al. Vulnerability assessment of OAuth implementations in Android applications [C] // Proc of the 31st Annual Computer Security Applications Conf. New York: ACM, 2015: 61-70

[33] Georgiev M, Iyengar S, Jana S, et al. The most dangerous code in the world: Validating SSL certificates in non-browser software [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2012: 38-49

[34] Fahl S, Harbach M, Muders T, et al. Why eve and mallory love android: An analysis of android SSL (in) security [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2012: 50-61

[35] Brubaker C, Jana S, Ray B, et al. Using frankencerts for automated adversarial testing of certificate validation in SSL/TLS implementations [C] //Proc of the 2014 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2014: 114-129

[36] Chau S Y, Chowdhury O, Hoque E, et al. SymCerts: Practical symbolic execution for exposing noncompliance in X.509 certificate validation implementations [C] //Proc of the 2017 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2017: 503-520

[37] Chen C, Tian C, Duan Z, et al. RFC-directed differential testing of certificate validation in SSL/TLS implementations [C] //Proc of the 40th Int Conf on Software Engineering. New York: ACM, 2018: 859-870

[38] Evans C, Palmer C, Sleevi R. IETF RFC 7469: Public key pinning extension for HTTP [DB/OL]. [2018-11-15]. https://tools.ietf.org/html/rfc7469

[39] Dukhovni V, Hardaker W. IETF RFC 7671: The DNS-based authentication of named entities (DANE) protocol: Updates and operational guidance [DB/OL]. [2018-11-15]. https://tools.ietf.org/html/rfc7671

[40] Szalachowski P, Matsumoto S, Perrig A. PoliCert: Secure and flexible TLS certificate management [C] //Proc of the 2014 ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2014: 406-417

[41] Laurie B, Langley A, Kasper E. Certificate transparency [DB/OL]. [2018-11-15]. https://www.certificate-transparency.org/

[42] Basin D, Cremers C, Kim H J, et al. ARPKI: Attack resilient public-key infrastructure [C] //Proc of the 2014 ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2014: 382-393

[43] Wendlandt D, Andersen D, Perrig A. Perspectives: Improving SSH-style host authentication with multi-path probing [C] //Proc of the 2008 USENIX Annual Technical Conf. Berkeley, CA: USENIX Association, 2009: 321-334

[44] Kasten J, Wustrow E, Halderman J A. CAge: Taming certificate authorities by inferring restricted scopes [G] // LNCS 7859: Proc of the Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2013: 329-337

[45] Soghoian C, Stamm S. Certified lies: Detecting and defeating government interception attacks against SSL (short paper) [C] //Proc of the Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2011: 250-259

[46] Gullasch D, Bangerter E, Krenn S. Cache games—Bringing access-based cache attacks on AES to practice [C] //Proc of the Security and Privacy. Piscataway, NJ: IEEE, 2011: 490-505

[47] Yarom Y, Falkner K. FLUSH + RELOAD: A high resolution, low noise, L3 cache side-channel attack [C] // Proc of the USENIX Security Symp. Berkeley, CA: USENIX Association, 2014: 22-25

[48] Liu F, Yarom Y, Ge Q, et al. Last-level cache side-channel attacks are practical [C] //Proc of the IEEE Symp on Security and Privacy. Piscataway, NJ, IEEE, 2015: 605-622

[49] Zhang Y, Juels A, Reiter M K, et al. Cross-VM side channels and their use to extract private keys [C] //Proc of the ACM Conf on Computer and Communications Security. New York: ACM, 2012: 305-316

[50] Disselkoen C, Kohlbrenner D, Porter L, et al. Prime + Abort: A timer-free high-precision L3 cache attack using Intel TSX [C] //Proc of 2017 USENIX Security. Berkeley, CA: USENIX Association, 2017

[51] Genkin D, Pachmanov L, Pipman I, et al. ECDSA key extraction from mobile devices via nonintrusive physical side channels [C] //Proc of the 2016 ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2016: 1626-1638

[52] Genkin D, Pachmanov L, Pipman I, et al. Stealing keys from PCs using a radio: Cheap electromagnetic attacks on windowed exponentiation [C] //Proc of the Int Workshop on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2015: 207-228

[53] Genkin D, Pipman I, Tromer E. Get your hands off my laptop: Physical side-channel key-extraction attacks on PCs [J]. Journal of Cryptographic Engineering, 2015, 5(2): 95-112

[54] Genkin D, Shamir A, Tromer E. RSA key extraction via low-bandwidth acoustic cryptanalysis [C] //Proc of the 34th Annual Cryptology Conf. Berlin: Springer, 2014: 444-461

[55] Halderman J A, Schoen S D, Heninger N, et al. Lest we remember: Cold-boot attacks on encryption keys [J]. Communications of the ACM, 2009, 52(5): 91-98

[56] Stewin P, Bystrov I. Understanding DMA malware [G] // LNCS 7591: Proc of the 9th Int Conf on Detection of Intrusions and Malware, and Vulnerability Assessment. Berlin: Springer, 2013: 21-41

[57] Blass E O, Robertson W. TRESOR-HUNT: Attacking CPU-bound encryption [C] //Proc of the 28th Annual Computer Security Applications Conf. New York: ACM, 2012: 71-78

[58] Harrison K, Xu S. Protecting cryptographic keys from memory disclosure attacks [C] //Proc of the 37th Annual IEEE/IFIP Int Conf on Dependable Systems and Networks. Piscataway, NJ: IEEE. 2007: 137-143

[59] Chow J, Pfaff B, Garfinkel T, et al. Understanding data lifetime via whole system simulation [C] //Proc of the USENIX Security Symp. Berkeley, CA: USENIX Association, 2004: 321-336

[60] Intel I. Software guard extensions programming reference, revision 2 [OL]. [2018-11-15]. https://software.intel.com/sites/default/files/managed/48/88/329298-002.pdf

[61] Müller U. Software grand exposure: SGX cache attacks are practical [C] //Proc of the 11th USENIX Workshop on Offensive Technologies. Berkeley, CA: USENIX Association, 2017

[62] Xu Y, Cui W, Peinado M. Controlled-channel attacks: Deterministic side channels for untrusted operating systems [C] //Proc of the 36th IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2015: 640-656

[63] Shinde S, Chua Z L, Narayanan V, et al. Preventing your faults from telling your secrets: Defenses against pigeonhole attacks [OL]. [2018-11-15]. https://arxiv.org/abs/1506.04832

[64] Lee J, Jang J, Jang Y, et al. Hacking in darkness: Return-oriented programming against secure enclaves [C] //Proc of 26th USENIX Security Symp. Berkeley, CA: USENIX Association, 2017

[65] Weichbrodt N, Kurmus A, Pietzuch P, et al. AsyncShock: Exploiting synchronisation bugs in Intel SGX enclaves [G] //LNCS 9878: Proc of the European Symp on Research in Computer Security. Berlin: Springer, 2016: 440-457

[66] Lipp M, Schwarz M, Gruss D, et al. Meltdown [EB/OL]. [2018-11-15]. https://arxiv.org/abs/1801.01207

[67] Kocher P, Genkin D, Gruss D, et al. Spectre attacks: Exploiting speculative execution [EB/OL]. [2018-11-15]. https://arxiv.org/abs/1801.01203

[68] Dan B, Ding X, Tsudik G, et al. A method for fast revocation of public key certificates and security capabilities [C] //Proc of the 10th USENIX Security Symp. Berkeley, CA: USENIX Association, 2001

[69] Ding X, Mozzacchi D, Tsudik G. Experimenting with server-aided signatures [C] //Proc of the Network and Distributed System Security Symp(NDSS 2002). Virginia: ISOC, 2002

[70] 林璟锵, 马原, 荆继武. 适用于云计算的基于SM2算法的签名及解密方法和系统. 中国发明专利 ZL2014104375995 [P]. 2017-11-03

[71] Jiang F, Cai Q, Guan L, et al. Enforcing access controls for the cryptographic cloud service invocation based on virtual machine introspection [C] //Proc of the 21st Int Conf on Information Security. Berlin: Springer, 2018: 213-230

[72] Perrig A, Perrig A, Perrig A. CASTLE: CA signing in a touch-less environment [C] //Proc of the 32nd Annual Conf on Computer Security Applications. New York: ACM, 2016: 546-557

林璟锵

博士，研究员，主要研究方向为应用密码学、网络与系统安全.

linjingqiang@iie.ac.cn

荆继武

研究员，主要研究方向为网络空间安全、身份管理与网络信任技术、系统安全理论与技术.

jing@is.ac.cn