来使浏览器只接受一些特定的证书;Google 提出并发布的数字证书透明性[7], 可以实时检测伪造证书, 出于谷歌的影响力, 这个方案的普及对于保障证书的安全性有显著的改善.

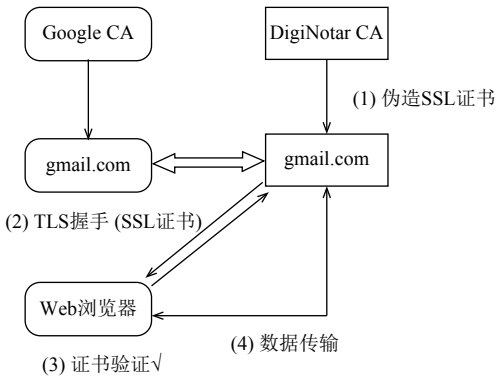

图 1 https 连接的建立过程 (中间人攻击)

2 基于 CT 的 Web PKI 基本原理

CT 的目标是提供一个开放透明的监控和审计系统, 要求 CA 向该系统中记录所有的证书签发行为, 从而让任何 CA 和域名所有者确定证书是否被错误签发或被恶意使用, 保护用户访问 https 网站时的安全.

CT 改变了证书的签发流程, 新流程规定: 证书必须记录到可公开验证、不可篡改且只能添加内容的日志中, 用户的 Web 浏览器才会将其视为有效. 通过要求将证书记录到这些公开的 CT 日志中, 任何感兴趣的相关方都可以查看由任何 CA 向任何网站签发的证书. 这能够促使 CA 在签发证书时更加负责, 从而有助于形成一个更可靠的系统.

CT 并不能阻止 CA 签发错误或虚假证书, 但是它能让人们清楚地看到 CA 签发的所有证书, 从而使检测这些证书的过程变得相对容易. 具体来说, CT 有三个主要的功能性目标: (1) CA 难以错发证书, 从源头上减少了错发证书地机率. (2) 提供一个公开的审计和监控系统. (3) 用户能够识别恶意/错误的证书.

如图 2 所示, CT 系统由三部分组成, 确保 CA 和日志服务器遵循 CT 工作流程:

(1) 证书 Log 日志: 维护可公开审计、只增不减的证书日志.

(2) 监控器: 通过下载并检查所有日志条目来检查日志中的可以证书.

(3) 审计器: 根据日志的部分视图验证日志的行为是否正确.

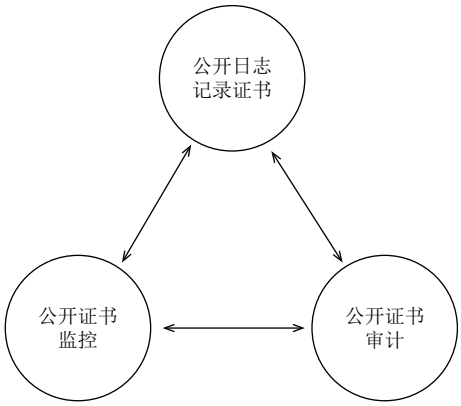

图 2 CT 系统组成部分

3 基于 CT 的 Web PKI 信任模型

基于 CT 的 Web PKI 信任模型分为带内带外两部分机制, 如图 3 所示. 其中带内信任机制是基于传统 Web-PKI 信任模型的, 证书颁发者 (CA) 是信任锚点, 而 Web 浏览器是负责验证证书的可信依赖方. 带外信任机制是基于 CT 的验证链路, 在此情况下, 依赖方不仅要信任传统 Web PKI, 还要信任 CT 机制中的 log 日志. 通过带内带外两种信任机制, 为整个 SSL 证书系统增加公共监督和审查功能来增强信任链模型.

3.1 基于传统 Web PKI 的带内信任模型

在传统 Web-PKI 信任模型中, 如图 4 所示, 浏览器预置了一组受信任的根证书. 可信的证书列表由浏览器供应商初始化、更新、在 TLS 连接建立期间, 目标网站提供相应域名的端点证书, 以及中间证书 (一个或多个), 旨在让浏览器构建从根到端点的证书信任链. 在这种情况下, 证书颁发者 (CA) 是信任锚点, 而 Web 浏览器是负责验证证书的可信依赖方[12]. 根据 X.5905 标准, 公钥需要通过数字证书绑定身份, 则依赖方可以确定 Web 服务器的真实性[13].

Web PKI 使用 CA 层次结构模型, 具体信任关系为: 如果证书由浏览器信任的 CA 签发, 则采用该证书的浏览器是可信的.Web PKI 有一组根 CA, 它们的公钥通常存储或预配置在根 CA 的受信任列表中、操作系统和浏览器中. 根 CA 作为整个 Web PKI 的基础, 证书路径是指从根 CA 证书到 Web 服务器证书的证书链. 链路验证是指验证路径的正确性和有效性.

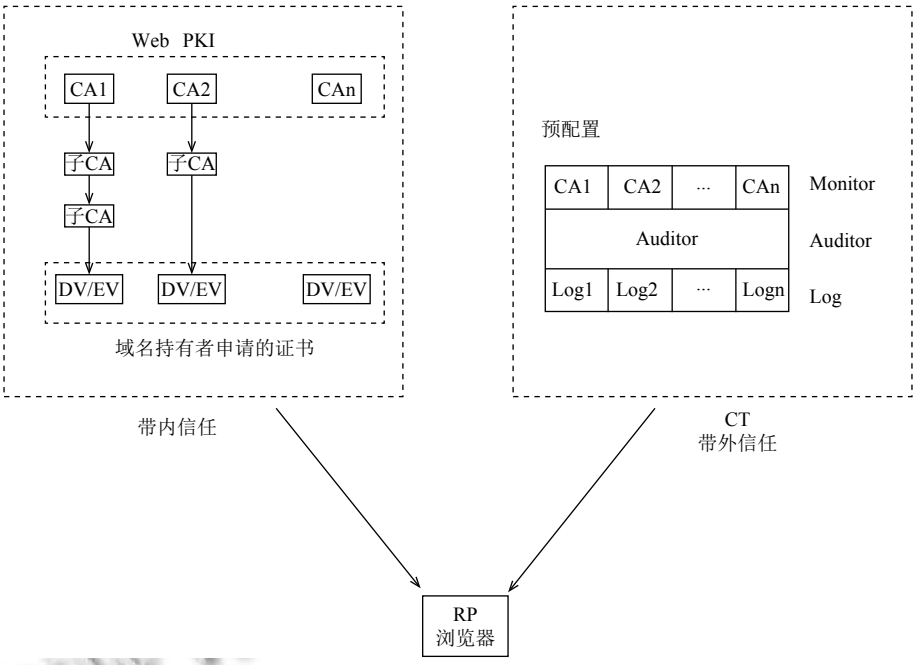

图 3 基于 CT 的 Web PKI 信任模型

对于依赖方浏览器来说, 为了确定公钥的合法性, 信任关系通过以下两个操作进行确认:

(1) 首先, 依赖方必须信任 CA 给签发证书的秘钥是合法的 (秘钥合法性).

(2) 其次, 依赖方必须信任 CA 颁发合法证书 (颁发者信任).

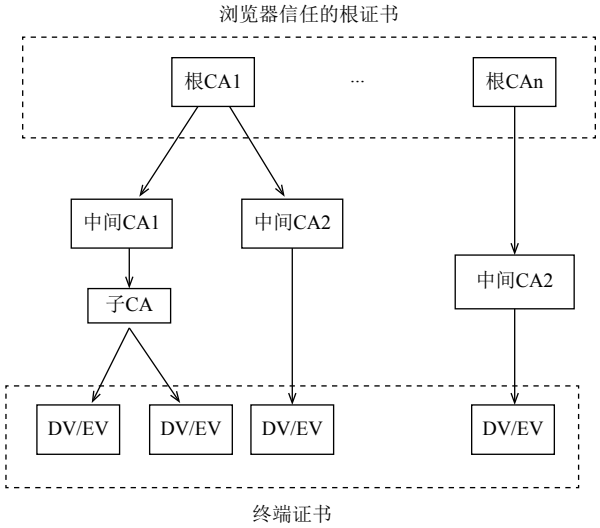

图 4 基于传统 Web PKI 的带内信任模型

在 Web PKI 中, 信任模型是颁发者信任和秘钥合法性二元信任. 由信任的根 CA 颁发的证书都是可信的, 但不同的 CA 可能实行不同的信任方案, 采用不同的安全机制.

但是, CA 可以任意颁发证书, 这使得 Web PKI 信任模型也存在中间人攻击的问题: 任何一个 CA 可以为任何一个网站签发证书, 无需该网站的同意. 拿到这种非授权的证书, 可以构造一个假冒的网站, 用户的浏览器在访问这种服务器时不会产生警告, 从而攻击者可以进行中间人攻击.

3.2 基于 CT 的带外信任模型

CT 改变了传统的 Web-PKI 信任模型, 除了要信任传统的 Web-PKI 模型, 即带内信任机制 CT, 还要信任 Log, Log 会对信任产生干预, 即带外信任机制. 基于 CT 的 Web PKI 信任模型中的带外信任机制主要由三个实体组成:

A. 证书日志 Log: 存储终端证书

B. 证书监控器 Monitor: 监控信任锚点

C. 证书审计器 Auditor: 浏览器验证证书

如图 5 所示, 基于 CT 的 Web-PKI 数据传输模型原理简述如下:

CA 在签发了某个 Web 服务器的证书之后, 及时地将该证书签发行为记录到公开的 Log 服务器上. Log 服务器以 Merkle Tree[14]的形式来存储 SSL 证书, 保持 Append-Only(只增不删) 特性. 即, 一旦 Log 接受了某个证书, 该 Log 的属性即可保证相应条目永远不会被移除或修改.

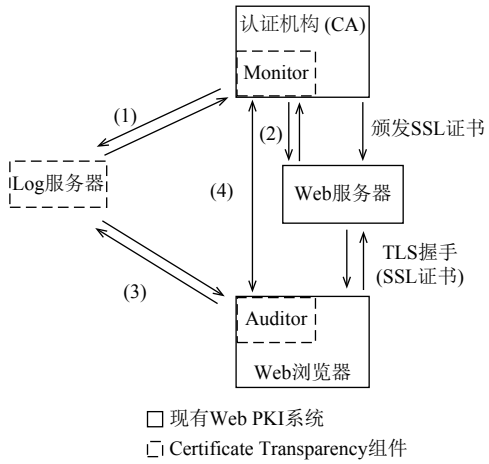

图 5 带外信任模型原理

更具体地说, CA 向 Log 服务器发送一个预签证书 (pre-certificate), Log 服务器使用自己的私钥签署一个证书签署时间戳 (Signed Certificate Timestamp, 简称 SCT), 并返回给 CA, 随后 CA 将 SCT 嵌入到正式的 SSL 证书中, 并发送给 Web 服务器. Web 服务器随后在 TLS 握手协议中将带有 SCT 签名的证书发送给 Web 浏览器.

域名管理者、CA 和利益第三方都可以部署 CT Monitor. 域名管理者通过 CT Monitor, 周期性的对 Log 进行监视, 可以实时得知自己的各个域名被 (部署了 CT 的所有 CA) 签发的证书, 并从中排查出可疑证书. 而 CA 也可以通过 CT Monitor 监视自己或其他 CA 签发的证书. 从而 CA 或域名管理者可以防止错误证书 (如伪造的服务器证书或未得到合法授权的中间证书) 被他人滥用.

Auditor 使用证书上附带的 SCT 签名来向 Log 服务器验证该证书是否被记录, 如果没有, Web 浏览器就可以拒绝访问该证书所对应的网站, 以保护自己的安全; 另一方面, Auditor 可以通过 CT 的 gossip 协议[15]将该问题证书的信息通知给 Monitor. 以便 CA 或域名管理者及时处理.

此外, 证书透明性中的 Gossip 协议[16]作为其通信协议, 允许 Monitor、Auditor、Web 客户端之间相互交流信息, 共享从 Log 服务器中获得的证书信息, 以保证证书的一致性、检测 Log 服务器的不正当行为.

3.3 信任核心

从 3.1 和 3.2 节分析可以看出, 浏览器 (依赖方 RP) 在基于 CT 的 Web-PKI 体系中是关键信任对象, 是连接传统信任模型和 CT Log 服务器的桥梁, 也是信任的核心. 它是带内带外两条数据验证的信任核心:

(1) 带内数据验证:

作为证书依赖方, SSL/TLS 协议客户端 (通常是浏览器) 预先存储有自己所信任的根 CA 自签名证书, 用来验证与之通信的 PKI 用户的证书链, 可信地获得该用户的公钥, 从而用于机密性、数据完整性、身份鉴别等各种安全功能. 浏览器选择哪些全球公开的信任锚点根证书, 也决定了后续有哪些 CA 是可信的.

(2) 带外数据验证:

由于 CT 机制的部署, 浏览器只接受放入 Log 服务器并由其信任背书的证书, 预示着浏览器在信任带内数据验证起点之前首先需要信任 Log 服务器的公钥.

综合这两个数据验证路径, 浏览器直接影响基于 CT 的 Web-PKI 系统中证书条目的有效性. 依赖方 RP 和 Auditor 分别是带内数据验证和带外数据验证的核心, 由此可见, 部署 CT 以后的整个 Web-PKI 体系信任核心是浏览器.

4 基于 CT 的 Web PKI 安全威胁模型

数字证书透明性 (CT) 这一概念及其相关技术体系, 是一次针对 Web PKI 安全缺陷在协议层面所做的系统性修复工作. 这一安全技术扩展, 改变了传统 Web PKI 的威胁模型, 但也引入了新的运行风险. 本节面向 CT 体系中的实体对象 (CA、证书、日志、监视器、浏览器等), 分析了“证书错误发放”错误类型和威胁因素, 提出了基于 CT 的 Web PKI 的安全威胁模型.

IETF 草案“Attack Model for Certificate Transprency”[17]讨论了 CT 在 Web PKI 背景环境中的威胁、潜在攻击场景. 现将草案中提到的威胁情况根据 CT 中的实体进行提炼概括为安全威胁模型, 如图 6 所示.

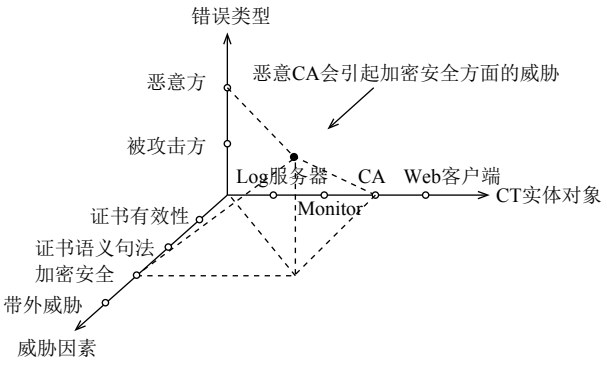

图 6 基于 CT 的 Web PKI 安全威胁模型

请注意, CT 中的实体 (Log 服务器、Monitor、CA、Web 客户端), 可能存在两种情况: 一种是实体本身为恶意“攻击者”, 产生一系列的威胁因素; 另一种可能性是, 实体是非恶意的, 由于操作错误或受到攻击的情况下, 产生一系列威胁因素.

在本模型中, 可以将威胁因素如下归类:

CT 实体	威胁因素
行为不正确的 Log 服务器	(a1): 为虚假证书创建条目 (a2): 向实体呈现不同的 Log 条目视图 (a3): 对于证书条目不进行语义检查 (a4): 将语义正确的证书报告为语义错误 (a5): 为虚假证书签发 SCT
行为不正确的 Monitor	(b1): 不通知从 CA 获得的证书中缺少 SCT (b2): 不通知域名所有者存在针对其域名的虚假证书 (b3): 向目标域名所有者发出虚假警告 (b4): 不报告证书存在语法错误 (b5): 将语义正确的证书报告为语义错误
证书颁发机构 (CA)	(c1): 未能将证书添加到日志中 (c2): 颁发错误的证书, 导致日志不执行检查 (c3): 一旦发现证书错发或被告知证书错发, 拒绝撤销或拖延撤销证书 (c4): 利用多个证书链, 导致虚假证书仍被信任
Web 客户端	(d1): 不拒绝没有 SCT 的证书

1) (a3, a5, b4, b5 和 c2) 这几类威胁涉及证书的有效性和语义验证.

2) 从密码学角度分析, 主要存在的威胁是 (a2 和 b3), 涉及的实体主要是 Monitor 和 Auditor. (通常不会将 Monitor 和 Auditor 分开讨论, 因为两者可以通过 Gossip 协议共同协作验证日志行为)

3) (c1 和 d1) 类为无法根除的带外威胁, 因为无法强制 CA 将证书添加到日志中, 也不能强迫客户端拒绝特定的证书. 但是如果客户端拒绝缺少 SCT 的证书, 那么违反 c1 类威胁的提交者会发现客户端拒绝了他们的证书.

4.1 行为不正确的 Log 服务器

CT 通过将证书添加日志, 从而达到公开审计的目的, 但若日志服务器是恶意的, 可能采取多种恶意行为. 具体如下:

日志可能假意阻止虚假证书的日志条目, 或者可以为虚假证书创建证书的条目, 但不报告其虚假性. 还可能会为虚假证书签署用于验证有效性的证书时间戳 SCT, 从而虚假证书会被视为有效.

此外, 如果行为不当的日志可能会向实体呈现不同的 Log 日志视图, 以帮助隐藏目标用户的虚假证书. 日志不会执行句法或语义检查, 或者简单地不发送错误报告. 这种情况, 我们无法通过 CT 机制审计证书.

另一种情况是, 恶意日志可能会把语法有效的证书报告成语法错误. 这可能导致 CA 进行不必要的调查工作, 或者可能不正确地撤销并重新颁发证书.

4.2 行为不正确的 Monitor

监控器监控日志中的可疑证书, 以发现非法或未经授权的证书、证书异常扩展或具有非法权限的证书. 监控器只要检测到异常情况, 就会通知用户.

若监视器是恶意的, 将不会执行句法或语义检查, 或者简单地不发送错误报告. 还可能会把语法有效的证书报告成语法错误, 这可能导致 CA 进行不必要的调查工作, 或者可能不正确地撤销并重新颁发证书.

更严重的是, 行为不正确的 Monitor 不会告知目标主体证书是虚假的, 甚至向目标域名所有者发出虚假警告.

4.3 证书认证机构 (CA)

在基于 CT 的 Web PKI 体系中, CA 的权力虽然被 Log Server、证书持有者分散, 但是仍存在安全隐患.

CA 若其选择不将证书添加到日志中, 从而屏蔽 CT 的监控作用; 若颁发虚假证书则可以伪装成经过公开审计的有效证书; CA 可能会拒绝撤销或拖延证书撤销的时间, 从而允许虚假证书仍存在一段时间, 从而引发安全隐患.

另一种情况, 恶意 CA 可能会反其道行之赢得信任: 通过撤销虚假的证书, 或列出虚假证书黑名单, 以避免浏览器厂商对 CA 采取惩罚措施.

更严重的是, 若多个父级 CA 为恶意的中间 CA 签发证书, 则恶意证书存在多个有效路径验证链, 即使其中一条链路验证被破坏, 其他链路验证有效, 如图 7 所

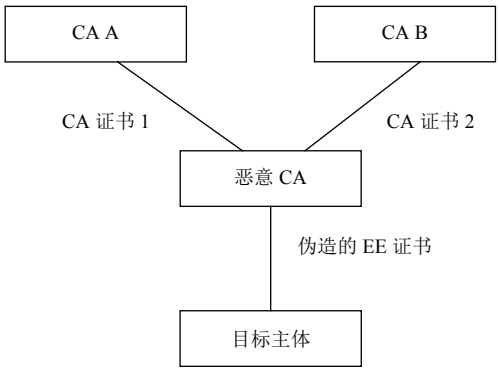

图 7 虚假证书的多个证书链

4.4 Web 客户端

在 CT 架构中, 必须对证书进行日志记录, 以使监视器能够检测到证书错发, 并触发后续撤销. 若浏览器拒绝没有 SCT 的证书、认为其无效, CA 为了使其颁发的证书有效, 会主动将其添加 Log 服务器, 进行日志记录, 以获得 SCT. 然而, 并不能强制浏览器拒绝缺少 SCT 证书, 因此这也是一种威胁因素.

5 基于 CT 的 Web PKI 安全保障机制

CT 技术开始在 Chrome 和 Chromium 中广泛部署后, 也引起了学术界的高度关注和后续研究. 文献[18,19]完成了对证书撤销操作的公开审计, 使得能在 Log 服务器上维护可审计的证书撤销状态信息. PoliCert[20]方案和 ARPKI[21]方案进一步限制了 CA 在 SSL/TLS 服务器证书签发过程中的权力. 文献[14]和[22]提出了更有效的 Gossip 协议, 确保 CT 用户获得的日志具有一致性, 也在 Gossip 协议中更加注重隐私保护. 文献[15]改进了 CT 技术使其具有扩展功能和短日志证明 (Proof) 格式.

通过上述文献分析, 对于 CT 技术方面的安全保障机制, 可以归结为以下两点:

(1)TLS PKI 基础架构中 CA 拥有很大权利, 域所有者无法控制其证书的使用和验证. CT 及其扩展增强方案, 都是在证书签发阶段, 由 Log 服务器、证书持有者借入来分散部分权利. 然而, 由于 CT 的目标仅仅是使证书公开审计, 但是攻击者仍可以入侵 CA 完成虚假证书的审计过程. 所以我们需要新的技术模型实现对 CA 权力进行限制的可信模型.

(2) 另外, CT 本身并没有对证书撤销进行改进, 所以技术方面需要改进证书撤销机制, 实现对证书撤销操作的公开审计. 证书撤销也必须能够验证一致性, 并且允许删除以及添加撤销状态. 证书撤销还必须能够有效地验证条目是否存在于日志的特定的版本中.

6 总结与展望

针对数字证书透明性的安全威胁问题, 本文首先梳理了基于 CT 的 WEB-PKI 的信任模型, 分析了信任模型的信任核心, 针对信任核心以及 CT 技术的其他实体对象归纳总结出安全威胁模型, 并提出了技术上的 ji 简易部署.

在 Web PKI 体系中, 认证机构 CA 给予的信任水平一直在下降, 基于日志方法的 CT 能够确保对 TLS 欺诈证书公开审计. 但针对本文安全威胁模型, 如何对于 CT 技术中 Log 服务器、Auditor、Monitor 等实体的不当行为进行检测和传播是目前基于日志的 PKI 体系结构的缺失; 并且对于 CT 传播过程中存在的隐私问题, 效率和可部署性都是具有挑战的研究方向. 近 2 年来, CT 技术在走向大面积推广的过程中, 其技术研究出现了新的深度. 我们有理由相信作为普适性的证书验证技术 CT 必定会有更加丰硕的研究成果. CT 技术的安全性部署、隐私性、对 Log 服务器的检测等工作将会在未来有更进一步的深入探讨和研究.

参考文献

1 Georgiev M, Iyengar S, Jana S, *et al*. The most dangerous code in the world: Validating SSL certificates in non-browser software. Proceedings of the 2012 ACM Conference on Computer and Communications Security. Raleigh, NC, USA. 2012. 38–49. [doi: 10.1145/2382196.2382204]

2 Al-Bassam M. SCPKI: A smart contract-based PKI and identity system. Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts. Abu Dhabi, United Arab Emirates. 2017. 35–40. [doi: 10.1145/3055518.3055530]

3 Amann B, Sommer R, Vallentin M, *et al*. No attack necessary: The surprising dynamics of SSL trust relationships. Proceedings of the 29th Annual Computer Security Applications Conference. New Orleans, LA, USA. 2013. 179–188.

4 Syta E, Tamas I, Visher D, *et al*. Certificate cothority: Towards trustworthy collective CAs. Proceedings of the 8th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs). 2015. 7.

5 Kumar D, Wang ZP, Hyder M, *et al*. Tracking certificate misissuance in the wild. Proceedings of 2018 IEEE Symposium on Security and Privacy (SP). San Francisco, CA, USA. 2018. 785–798.

6 林璟锵, 荆继武, 张琼露, 等. PKI 技术的近年研究综述. 密码学报, 2015, 2(6): 487–496.

7 Laurie B, Langley A, Kasper E. Certificate transparency. https://datatracker.ietf.org/doc/rfc6962. [2015-10-14].

8 Dowling B, Günther F, Herath U, *et al*. Secure logging schemes and certificate transparency. In: Askoxylakis I, Ioannidis S, Katsikas S, et al, eds. Computer Security – ESORICS 2016. Cham: Springer, 2016. 140–158, doi: 10.1007/978-3-319-45741-3_8.

9 Hodges J, Jackson C, Barth A. HTTP strict transport security. http://www.rfc-editor.org/info/rfc6797.
10 Evans C, Palmer C, Sleevi R. Public key pinning extension for HTTP. https://datatracker.ietf.org/doc/rfc7469. [2015-10-14].
11 Hoffman P, Schlyter J. The DNS-based authentication of named entities (DANE) transport layer security (TLS) protocol: TLSA. https://datatracker.ietf.org/doc/rfc6698. [2015-10-14].
12 Delignat-Lavaud A, Abadi M, Birrell A, *et al*. Web PKI: Closing the gap between guidelines and practices. Proceedings of 2014 Network and Distributed System Security Symposium. San Diego, CA, USA. 2014.
13 Braun J, Volk F, Buchmann J, *et al*. Trust Views for the Web PKI. Proceedings of the 10th European Workshop on Public Key Infrastructures, Services and Applications. Egham, UK. 2013. 134–151.
14 Eskandarian S, Messeri E, Bonneau J, *et al*. Certificate transparency with privacy. Proceedings on Privacy Enhancing Technologies, 2017, 2017(4): 329–344. [doi: 10.1515/popets-2017-0052]
15 Singh A, Sengupta B, Ruj S. Certificate transparency with enhancements and short proofs. In: Pieprzyk J, Suriadi S, eds. Information Security and Privacy. Cham: Springer, 2017. [doi: 10.1007/978-3-319-59870-3_22]
16 Nordberg L, Gillmor DK, Ritter T. Gossiping in CT. https://datatracker.ietf.org/doc/draft-ietf-trans-gossip. [2018-01-14].
17 Kent S. Attack and threat model for Certificate Transparency. https://tools.ietf.org/html/draft-ietf-trans-threat-analysis-12. [2017-10-13].
18 Ryan MD. Enhanced certificate transparency and end-to-end encrypted mail. Proceedings of 2014 Network and Distributed System Security Symposium. San Diego, CA, USA. 2014. [doi: 10.14722/ndss.2014.23379]
19 Laurie B, Kasper E. Revocation transparency. Google Research, 2012.
20 Szalachowski P, Matsumoto S, Perrig A. PoliCert: Secure and flexible TLS certificate management. Proceedings of 2014 ACM SIGSAC Conference on Computer and Communications Security. Scottsdale, AZ, USA. 2014. 406–417. [doi: 10.1145/2660267.2660355]
21 Basin D, Cremers C, Kim THJ, *et al*. ARPKI: Attack resilient public-key infrastructure. Proceedings of 2014 ACM SIGSAC Conference on Computer and Communications Security. Scottsdale, AZ, USA. 2014. 382–393. [doi: 10.1145/2660267.2660298]
22 Chuat L, Szalachowski P, Perrig A, *et al*. Efficient gossip protocols for verifying the consistency of Certificate logs. Proceedings of 2015 IEEE Conference on Communications and Network Security (CNS). Florence, Italy, 2015, 391(403). [doi: 10.1109/CNS.2015.7346853]