Limitations of IEC62351-3's Public Key Management

James G. Wright
School of Mathematics and Information Security,
Royal Holloway, University of London,
Egham TW20 0EX,
United Kingdom
Email: james.wright.2015@live.rhul.ac.uk

Stephen D. Wolthusen
School of Mathematics and Information Security,
Royal Holloway, University of London,
Egham TW20 0EX,
United Kingdom
Email: stephen.wolthusen@rhul.ac.uk

***Abstract*—The ISO/IEC 62351 standard provides a set of security controls and protocols for communications in smart grids based on the ISO/IEC 60870, 61850, and DNP3 standards. It offers the protection goals of confidentiality, integrity, and authentication. In this paper we perform a systematic study of the ISO/IEC 62351-3 standard regarding the use of public key infrastructure in smart grid communication. We show that the standard at present does not align with the quality of service requirements for performance and interoperability in the ISO/IEC 61850 standard and thereby may jeopardise effective operations. We demonstrate that it is possible to claim conformance with the ISO/IEC 62351-3 standard but be vulnerable to denial of service attacks arising from insufficiently specified behaviour for public key certificate validation and revocation. Further issues can give rise to downgrade attacks against cipher suites and protocols used, allowing a man-in-the-middle attacks contrary to the standard's claims.**

I. Introduction

The smart devices that are currently being integrated into power systems are degrading the ''air gap" security principle that has been used by the sector for the past few decades. The guiding principle of security through obscurity of supervisory control and data acquisition (SCADA) protocols is no longer tenable, as their communications networks are beginning to interact with internet technologies. Whilst both the academic and industrial research communities are now focusing on solving the unique security challenges faced by smart grids, there is very little focus dedicated to checking if the security promises made by the various protocols used in the space hold true. Securing the protocols could prevent some of the theorised attacks against smart grids.

The following analysis builds upon previous work[29] and looks at the security promises made by IEC62351-3[20]. This section of the IEC62351 is designed to encrypt the data packets being sent across a communications network that use either IEC60870, IEC61850, or DNP3 substation automation standards. It adds the security promises of confidentiality, integrity, and message level authentication to the communications protocols it is deployed on. The protocol describes how smart grid TCP/IP communication networks should be secured via public key infrastructure. The kinds of communications that it will be applied to are those that don't need a real time response, such as sampled values (SV) in IEC61850. IEC62351-6, the section that describes how IEC61850 will be secured, recommends that ''*applications using GOOSE and IEC 61850-9-2 and requiring 4msec response times, multicast configurations, and low CPU overhead*" should not use encryption[21]. IEC62351-3 claims that it will counter man-in-the-middle and replay attacks, along with negating any eavesdropping attempts. The key contribution of this paper is to show that these promises are not upheld if certain policy omission regarding the certificate authority (CA) trust networks aren't fixed, which could allow attackers to implement various denial of service attacks using using the system's logic whilst still being compliant with the protocol.

It is shown that the approach in IEC62351-3 of just declaring the use transport layer security(TLS), message authentication codes(MAC), and CAs as a way of fulfilling its promises still leaves the network vulnerable, as it neglects to comment on CA implementation, certificate validation and revocation policies. As well undermining its security promises, it is shown that these omissions come into clash with the IEC61850-5 quality of service requirement of interoperability of devices[22].

The remainder of this paper proceeds as follows: Section II describes the work currently being done on the security of IEC62351. Section III briefly describes how IEC62351-3 implements TLS, and its supporting standards. Section IV then describes omissions that could be used in attacks against the trust network. Section V describes how downgrade attacks are possible in the current standard, before giving a conclusions and a direction for future work in section VI.

II. Related works

Most of the research into IEC62351 security is usually part of an analysis into the state of either general smart grid/SCADA, or IEC61850 security [28] [11] [9]. It is usually presented as a solution to the taxonomy of problems that the authors have identified in their reviews.

However, there is some research that looks at the challenges faced in implementing IEC62351. Tawde et. al.[24] propose a bump in the wire key management mechanism to implement IEC62351-5, which is designed to extend the security promises of IEC60870 and DNP3. They propose connecting the bump in the wire devices to the remote terminal units and management terminal units on the network topology. They claim that this would be a practical way of bringing legacy hardware in line with the standard, but make no consideration as to whether the added latency will violate the quality of service requirements of IEC60870 or DNP3.

Some work has been done on how IEC62351-4 provides security for the manufacturing message specification (MMS) model which is used in IEC61850 and IEC60870. Fries et. al.[11] identify that MMS messages that use multiple transport layer connections undermine the promise of integrity, as it is assumed that an intermediary in the chain is trusted. They propose introducing security sessions into the MMS protocol to restore this promise. Ruland & Sassmannshausen[16] propose adding the security promises of non-repudiation and traceability to IEC62351-4 to enhance MMS security. They argue that the authentication of communication partner does not guarantee authentication of the origin of data transmitted. They propose encoding XML non-repudiation tokens into the data sent between the client and the server. These tokens will contain proof of authorship and a timestamp.

Some research has been done on how IEC62351-3's public key infrastructure key management schemes could be implemented. Zhao et. al.[30] points out that certificate revocation is not actually part of IEC 62351-3, and propose using a broadcast encryption media key block to secure the communications in a hierarchical device structure which would allow them to revoke a device's certificate. However, they provide no proof as to if their algorithm actually meets the security promises of IEC62351-3. Fuloria et. al.[12] discuss the various possible encryption choices, and their implementations, given the limited computational resources the communications network will have. They also develop a broad threat model with which encryption can defend against. However, their analysis overlooks what could potentially go wrong in key update algorithms. They conclude that encryption may be too great a burden to implement on smart grid network.

No further works have been found on IEC62351-3 security. The purpose of the following analysis is to show that if no decision is made on the public key infrastructure's implementation of certificate revocation update frequency and processing overhead, then, regardless of the chosen key management topology, the protocols security promises will not be upheld.

III. IEC62351-3's Implementation of TLS and Public Key Infrastructure

As it currently stands the IEC62351-3 deployment of public key infrastructure defers to the TLS standard, and its dependencies, on how it should be implemented across its trust architecture. The documentation states that it will support the current version of TLS, which is predominately TLS 1.2, as well as any subsequent iterations, such as the under development TLS 1.3[15]. However, it also allows for backwards compatibility up to TLS 1.0. Assuming that the implementer of this trust architecture uses TLS 1.2, the possible features used in its deployment shall be derived below.

The TLS implementation used must support bi-directional certificate authentication of each party in a session. This means any implementation of the TLS handshake algorithm used on the smart grid's communications network should have the server send a 'CertificateRequest' message before it sends its 'ServerHelloDone' message. When the client is initiating the handshake protocol with a server, they define a list of cipher suites that they wish to use in the following session. The only requirement that IEC62351-3 definitively makes is that a ''*cipher suite that specifies NULL for encryption shall not be used for communications*"[20]. The only other assertion it makes is it would prefer the manufactures to use a SHA cipher suite and avoid MD5. Whilst TLS 1.3 removes support for MD5 and SHA-224[15], IEC6235-3 makes no references to the documentation that updates the list of cipher suites TLS 1.2 can use. It explicitly points to the ancillary standards regarding the renegotiation extension and the dropping of support of SSL 2.0, but overlooks the one that revokes the use of RC4 suite[14]. If that is not included, RC4 becomes an acceptable standard to use in the implementation which would make the TLS implementation vulnerable to historically known attack vectors.

For the TLS handshake algorithm to work there has to be some form of public key infrastructure in place to allow parties to authenticate each others identities. The current implemention of a CA's certificate revocation lists (CRL) on the trust architecture that TLS 1.2 use is described in the RFC5280[5] standard. RFC5280 states that entity connected to a certificate can either be an individual or a computational device. RFC5280 is also the standard that IEC62351-3 defers to for its CRL infrastructure. However, this standard does not present any algorithm for how a device can validate a certificate it receives.

The documentation for IEC62351-3 suggests that the facilitation of the smart grid's public key infrastructure shall be done either by a network of localised CAs storing a CRL, or some form of the Online Certificate Status Protocol (OCSP)[20]. For the administration of the networks CRL's IEC62351-3 defers to best practices of key management that are laid out in IEC9594-8[25] and IEC62351-9[23], which is currently under development. IEC9594-8 provides some considerations that should be made when a trust architecture is first set up. It makes no concrete declarations on what an implementer has to do. For CRL implementations it provides consideration for the deployment of trust anchors, certification paths, key generation, and CRLs. Whilst it acknowledges that OCSP is an alternative method of certificate validation to the distribution of a CRL, it does not make any considerations on its implementation. Whilst IEC62351-3 states that an OCSP algorithm can be used as part of its public key infrastructure, it makes no reference to a specific implementation. Inferring from IEC9594-8 it could be assumed that it is referring to RFC6960[17], but it could also be an OCSP stapling implementation described in RFC6066[7], which is referenced by IEC6235-3. Whilst both OCSP implementations differ in the way they distribute the onus of processing the OCSP request, they both depend on having a secured and honest CA server on the trust architecture to manage a CRL. Regardless of the certificate revocation mechanism that is implemented it must be noted that any requests sent to a CA for information regarding a certificates validity is transmitted via an unencrypted channel.

IV. IEC62351-3 Policy Omissions Regarding Public Key Infrastructure

Whilst IEC62351-3 does go some way to securing the communication networks deployed using IEC60870, IEC61850, and DNP3, it overlooks certain key areas that would allow it to fulfil its promises. IEC62351-3 secures the data packets transmitted using the TLS protocol, but that depends upon the proper implementation of public key infrastructure. The standard infers that it will deploy either a CRL or an OCSP algorithm. However both IEC62351-3 and TLS abdicate responsibility on the best practise for the operation of a CA. Whilst IEC62351-3 does say it will deploy CAs within the network, it omits to lay out any common policy on how they should be operated. If these omissions persists then it could undermine IEC61850-5's quality of service requirement of interoperability of devices on the network, as well as its own security promises. It is the concern of the authors that without a proper administrative policy on how the CAs are deployed, then an attacker could circumvent the protection provided by TLS.

Shown below are the possible problems that can arise with the major implementations of trust architecture, CRLs, OCSP, and OCSP stapling. Potential solutions to these problems are suggested. After that, the overview turns to the specific considerations that need to be taken into account when public key infrastructure is deployed on a smart grid's communication network.

A. Problems with CRL's

Below are potential attack vectors that can arise when using CRL based trust architecture:

- Public key infrastructure is only as robust as the topology of the trust architecture that it is deployed on[26]. The correct choice of trust topology can help provide a greater guarantee that a device on the network can authenticate the certificates it receives in a TLS handshake. For example, if the trust topology implemented is a hierarchical one, then an attacker could mount a denial of service against the promise of authentication. They could deny a targeted device access to the CA's available on its branch, compromising it's availability to the grid.
- Owners of trust infrastructure need some way of checking that a CA on the network is legitimate, and should have the authority to issue certificates. If the CA cannot be trusted to act honestly when checking the veracity of the identity of an entity requesting a certificate, and for them to maintain an accurate CRL, then the promises of both authentication and integrity cannot be upheld.[10]
  Although there haven't been any reported cases where a CA has acted in a intentionally dishonest manner we can deduce the consequences of not validating their honesty by looking at examples of CA error from recent history[2]. In 2013 Türktrust accidentally marked two customer certificates as CA certificates. Someone then used one these on a network gateway, which allowed them to intercept and decrypt traffic leaving it. In 2012 TrustWave issued an intermediate CA certificate to a customer that then used it to generate end-user certificates, that allowed them to decrypt traffic. In 2011 both Comodo and DigiNotar CAs networks were compromised. In the first instance the attackers issued themselves nine certificates for popular domains, however only one of them was seen deployed in the wild. In the latter incident, the malicious certificates were used to implement a man in the middle attack, an attack which IEC62351-3 says it prevents, against Iranian users of Gmail.
  The above cases show that if the CA cannot be trusted to perform its signing function honestly, then it compromises the global trust architecture.
- Certificates used in a trust network should only be accepted at the beginning of a session, if it can be shown that it has come from a trusted CA. If a client is unaware of how a CA acts when it receives a certificate signing request, then they are uncertain in the level of trust they can place in any certificates it receives. If anyone can get a certificate for their device without any check that the person has permission to make a certificate signing request, then the promise of authentication cannot be upheld. If there is no validation by the CA of a requester, then an impersonation attack[4] can take place. This is further complicated when the party requesting the CA to sign a certificate is a device rather than an individual. This means that there is no guarantee that a device on the network is sending its data to a legitimate party.
- CRL requests are sent unencrypted across the communications network. This problem could allow an attacker to violate the promises of confidentiality, integrity and authentication by manipulating or intercepting packets, via a man in the middle attack. Due to this weakness in the CRL algorithm, it is possible for an attacker to mount a denial of service attack against a device. As the CRL response has no boundary on its length[5], an attacker could reply a device's request for an updated CRL with a packet that far exceeds the overhead it can dedicate to the task of processing.

B. Problems with the OCSP algorithm

The OCSP algorithm addresses the final issue shown in section IV-A. By having fixed constant message complexity, as indicated by the responses content-length header[17], an attacker can no longer send a reply that would exceed a devices dedicated overhead. However, the OCSP documentation makes no comment on the optimal latency a device should allow

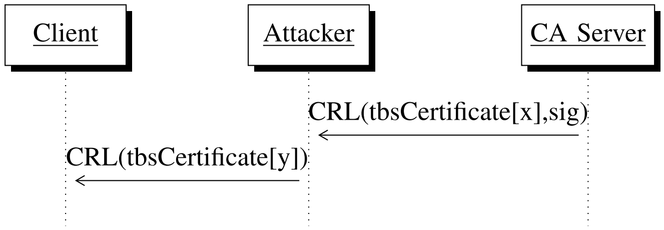

Fig. 1: The attacker sends an array of garbage data for the device to process. Where x and y are the number of bits in the array, and y ≫ x

when waiting for a reply. The behaviour of what a device should do if it receives no or an invalid reply can improve how effective it is at upholding its security promises. Without any guidance on this issue a device could be made to accept a revoked or expired certificate by an attacker, which would undermine the promises of integrity and authentication. As the OCSP response is not transmitted over an encrypted channel, like CRLs, an attacker can modify the response data packets. If the attacker changes the response to 'tryLater' the client doesn't require a signed response, and, depending on the implementation, the algorithm may 'soft fail' and accept the certificate[8].

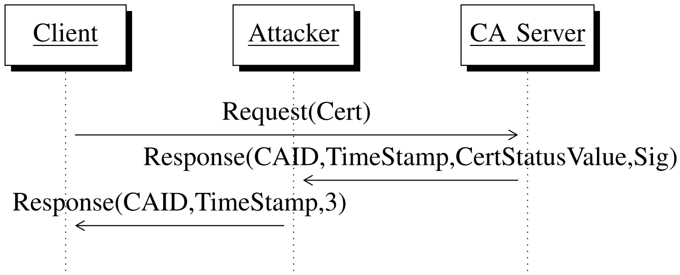
Client
Attacker
CA Server
Request(Cert)
Response(CAID,TimeStamp,CertStatusValue,Sig)
Response(CAID,TimeStamp,3)

Fig. 2: A session diagram of the 'tryLater' attack.

C. Problems with the OCSP stapling algorithm

OCSP stapling is designed to allow CA's to reduce the amount of overhead they dedicate to the requests they receive. It achieves this by allowing the certificate holder to query the CA's database itself to get a signed timestamped validation response. However the binding of the certificate and the CA's validation signature must be completely unalterable, otherwise it then leaves verification in the hands of an attacker. This would undermine any promise of authentication. Possible ways of achieving this binding are described in section IV-D.

Another problem with this validation methodology is there needs to be a redundancy in the trust architecture if the server is unable to get their certificate signed and timestamped by a CA it should no longer be available to communicate with as no device an authenticate its identity, which undermines the IEC61850-5 promise of availability.

D. Potential Solutions and Considerations

To secure the trust architecture of any communications network using IEC62351-3, the protocol would need to go beyond what is written in the referenced public key infrastructure specifications. To improve the security of the CA, the protocol would have to declare the best practices expected of any entity issuing certificates on the communication before allowing them access[19].

To prevent the 'soft fail' scenario of the 'tryLater' attack in OCSP, IEC6235-3 should make it clear that if a device receives a 'tryLater', revoked, or even no response that it should not continue the session with the other party, as there is no guarantee of authentication.

To make sure that the binding of a certificate and the CA's validation is unalterable in an OCSP stapling trust system, there are two possible ways of deploying it. The first method would be to create a dedicated private CA for the grid's communications network. This way an intelligent electronic device can reduce the overhead needed to check that the CA is a valid entity, as if it wasn't the private CA it would just end the session. For this approach to work the private CA would have to develop stringent policies on how it manages its private key(s), as if those are lost then the networks entire trust model would be compromised[13]. It should also be noted that most commercial CA lists used in web browsers are cultivated to be used in as many territories as possible, some of which may decide to use its trust infrastructure to perform man in the middle attacks against its own citizens. If one of these states CA's is used on the smart grid trust network, it could be used for similar purposes[18]. The other method is bind the DNSsec[6] of the communications network with the trust topology. By making a specific tier of the grid's DNS architecture the only CA for the tier immediately below it, a chain of trust can be built between tiers.

It must be noted that only one of the two previous suggestions should be implemented at a time. Using them in combination would allow the private CA to undermine the DNSsec chain of trust as it would allow CA's from any tier be accepted.

E. Considerations that need to be made for Smart Grid implementation

There are two challenges faced by any public key infrastructure that is implemented on a smart grid's communication network. The first is how frequently an intelligent electronic device should receive a copy of the latest CRL[27]. A device is vulnerable to communicating with a malicious server that has a presented an invalid certificate between the time it is revoked and when the device receives the new CRL. Most web CAs send out an updated CRL every seven days. This leaves a web browser potentially vulnerable for six days between updates. It is conceivable that the CRL wont be sent until it has reached a significant number of revocations. Having a definitive policy on the time between CRL updates would remove any uncertainty as to when a device will receive the latest CRL, which would reduce the chance that it will accept an invalid certificate.

The second problem that needs to be considered is the overhead an intelligent electronic device can suffer when processing a CRL request. The device must validate the CA's signature, perform databasing tasks every time it receives a CRL, and parse the database to check if the certificate it has received in a handshake is still valid. The device has limited processing power and storage to perform these tasks, and no consideration is made for the latency requirements in which these tasks must be completed. Without any quality of service requirement imposed on processing of any CRL request, the device could be susceptible to an attacker using revoked certificates until the CRL has been processed and parsed. However, IEC62351-3 does make some consideration on the storage of CRL's. It suggests that a CRL stored on an intelligent electronic device should be no larger than 8192 octets. Unfortunately it provides no suggested process if a CRL is larger than this. There would be no way of remedying an attack implemented along this vector, without external interference, as the protocol does not allow a checking or inability to access a CRL to end an established session[20].

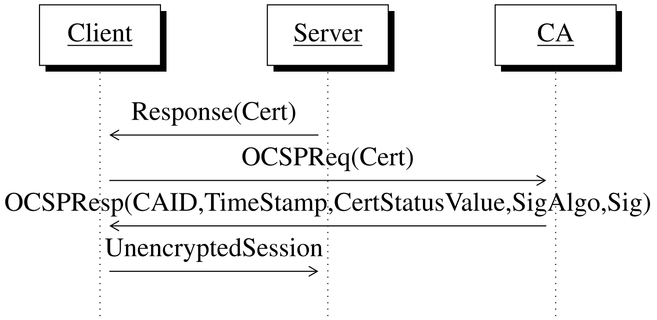

Fig. 3: An example of a session downgrade after the CA server uses an undesirable signing algorithm in its OCSP response.

V. IEC62351-3 Policy Omissions that could lead to Downgrade Attacks

Due to the accepted development practise of choosing interoperability of devices over security requirements, which is enshrined in IEC61850-5, IEC6235-3 cannot guarantee the confidentiality of data sent over its network.

The specification makes no comment on what happens if there is no agreement during a TLS handshake when a signature algorithm. The danger of having no policy for this eventuality means the manufacturer could make the judgement for their device to use on no encryption in this instance, which attacker could exploit in an attack. A potential justification for the manufacturer making this decision is it reduces the overhead on the processor of the intelligent electronic device, which would make it easier to be compliant with IEC62351. An example of a downgrade type attack was developed against TLS 1.2, as it was shown that a lot of distributions were being implemented with the export grade Diffie-Hellman 512 bit primes as a possible cipher suite. These could be precomputed[1], therefore allowing messages to be compromised in transit. It should be noted that IEC62351-3 wouldn't be susceptible to this attack as 1024 bit key length is the minimum requirement for key exchange.

IEC62351-3 should make explicitly clear what happens to a session if there is no agreement between parties on TLS protocol version or CA response. Whilst this comes into direct clash with the requirements of IEC61850-5, it would guarantee the integrity and confidentiality of the data being transmitted over the smart grid's network. As previously noted the protocol provides backwards compatibility up to TLS 1.0. It would also strengthen IEC62351-3 security promises to remove this requirement, as these have been shown to be insecure[3].

VI. Conclusion

The above overview shows that the omissions in IEC62351-3's trust architecture allows implementers to be fully compliant with the specification whilst undermining its security promises. This can be remedied by extending IEC62351-3 to include key management administration. It has also been discussed that there needs to be consideration made for what happens in case the trust infrastructure is attacked or damaged. Decisions on the preferred topology of the trust network or providing an algorithm on what an intelligent electronic device should do if it fails to receive an affirmative response from a CA server, can help the protocol uphold its security promises. Without declaring a key management policy an attacker can undermine the promises of message confidentiality, integrity, and authentication. This could allow them to send malicious commands that could lead to physical damage being inflicted upon the grid.

It was also shown that it's possible for a downgrade attack to be implemented due to the interoperability requirements of IEC61850-5. It should also be noted that the current state of IEC62351-3 would be a hindrance interoperability, as it leaves a lot of decision to the implementers and manufacturers of intelligent electronic devices about the deployment of the public key infrastructure. Without a consensus on how the network should be operated the only way the devices could possibly work together is at the lowest common denominator of security. Although this overview has been done by looking at how IEC62351 interacts with IEC61850, there is a reasonable likelihood that the quality of service requirements that have been used in the above analysis exist in other communications protocols, such as IEC60870 or DNP3.

Whilst it would be useful to be able to test the attacks developed above, there are currently no testbed systems with IEC62351-3 implementations. The authors plan to proceed by following on from this paper with performing a similar analysis on the other sections of IEC62351. Once a review of the other sections is completed, the focus shall move to formally proving the attacks theorised are actually possible by using context-free grammars methods developed in a previous work[29] to create a set of rules that prove that the promises are not upheld.

ACKNOWLEDGMENT

This work is supported by an EPSRC Academic Centres of Excellence in Cyber Security Research PhD grant.

REFERENCES

[1] D. Adrian, K. Bhargavan, Z. Durumeric, P. Gaudry, M. Green, A. J. Halderman, N. Heninger, D. Springall, E. Thomé, L. Valenta, B. VanderSloot, E. Wustrow, S. Zanella-Béguelin, and P. Zimmermann. Imperfect forward secrecy: How diffie-hellman fails in practice. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, pages 5–17, New York, NY, USA, 2015. ACM.

[2] B. Amann, R. Sommer, M. Vallentin, and S. Hall. No attack necessary: The surprising dynamics of ssl trust relationships. In *Proceedings of the 29th Annual Computer Security Applications Conference*, ACSAC '13, pages 179–188, New York, NY, USA, 2013. ACM.

[3] G. V. Bard. A challenging but feasible blockwise-adaptive chosen-plaintext attack on ssl. In *SECRYPT 2006, PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON SECURITY AND CRYPTOGRAPHY, SET'UBAL*, pages 7–10. INSTICC Press, 2006.

[4] P. Black and R. Layton. Be careful who you trust: Issues with the public key infrastructure. In *Proceedings of the 2014 Fifth Cybercrime and Trustworthy Computing Conference*, CTC '14, pages 12–21, Washington, DC, USA, 2014. IEEE Computer Society.

[5] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. RFC5280 - Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile. Technical report, Internet Engineering Task Force, 2008.

[6] D. Eastlake. RFC2535 - Domain name system security extensions. Technical report, Internet Engineering Task Force, 1999.

[7] D. Eastlake. RFC6066 - The transport layer security (TLS)extensions: Extension definitions. Technical report, Internet Engineering Task Force, 2011.

[8] W. El-Hajj. The most recent SSL security attacks: origins, implementation, evaluation, and suggested countermeasures. In *Security and Communication Networks*, volume 5, pages 113–124, 2012.

[9] A. Elgargouri, R. Virrankoski, and M. Elmusrati. IEC 61850 based smart grid security. In *Industrial Technology (ICIT), 2015 IEEE International Conference on*, pages 2461–2465, March 2015.

[10] C. Ellison and B. Schneier. Ten risks of PKI: What you're not being told about public key infrastructure. In *Computer Security Journal*, volume 16, pages 1–7, 2000.

[11] S. Fries, H. J. Hof, and M. Seewald. Enhancing IEC 62351 to improve security for energy automation in smart grid environments. In *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*, pages 135–142, May 2010.

[12] S. Fuloria, R. Anderson, F. Alvarez, and K. McGrath. Key management for substations: Symmetric keys, public keys or no keys? In *Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES*, pages 1–6, March 2011.

[13] S. Kent. Evaluating certification authority security. In *Aerospace Conference, 1998 IEEE*, volume 4, pages 319–327 vol.4, Mar 1998.

[14] A. Popov. RFC7465 - Prohibiting RC4 cipher suites. Technical report, Internet Engineering Task Force, 2015.

[15] E. Rescorla. The transport layer security (TLS) protocol version 1.3 draft-ietf-tls-tls13-11. Technical report, Internet Engineering Task Force, 2016.

[16] K. C. Ruland and J. Sassmannshausen. *Non-repudiation Services for the MMS Protocol of IEC 61850*, pages 70–85. Springer International Publishing, Cham, 2015.

[17] S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. RFC6960 - X.509 internet public key infrastructure online certificate status protocol - OCSP. Technical report, Internet Engineering Task Force, 2013.

[18] C. Soghoian and S. Stamm. *Certified Lies: Detecting and Defeating Government Interception Attacks against SSL (Short Paper)*, pages 250–259. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[19] Symantec. Prioritizing trust: Certificate authority best practices. https://www.symantec.com/content/en/us/enterprise/white_papers/b-prioritizing-trust-ca-best-practices_WP.en-us.pdf.

[20] TC 57 Power systems management and associated information exchange. Power systems management and associated information exchange, data and communication security - part 3: Communication network and system security - profiles including TCP/IP. IEC standard 62351-3. Technical report, International Electrotechnical Commission, 2007.

[21] TC 57 Power systems management and associated information exchange. Power systems management and associated information exchange, data and communication security - part 6: Security for iec 61850 . IEC standard 62351-6. Technical report, International Electrotechnical Commission, 2007.

[22] TC 57 Power systems management and associated information exchange. Communication networks and systems for power utility automation - part 5: Communication requirements for functions and device models. IEC standard 61850-5. Technical report, International Electrotechnical Commission, 2013.

[23] TC 57 Power systems management and associated information exchange. Power systems management and associated information exchange, data and communication security - part 9: Cyber security key management for power system equipment. IEC standard 62351-3. *Currently under development*. Technical report, International Electrotechnical Commission, 2017.

[24] R. Tawde, A. Nivangune, and M. Sankhe. Cyber security in smart grid SCADA automation systems. In *Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015 International Conference on*, pages 1–5, March 2015.

[25] JTC 1/SC 6 Telecommunications and information exchange between systems. Information technology – open systems interconnection – the directory – part 8: Public-key and attribute certificate frameworks. IEC standard 9594-8. Technical report, International Electrotechnical Commission, 2014.

[26] Z. E. Uahhabi and H. E. Bakkali. A comparative study of PKI trust models. In *2014 International Conference on Next Generation Networks and Services (NGNS)*, pages 255–261, May 2014.

[27] S. Vandeven. Digital certificate revocation. https://www.sans.org/reading-room/whitepapers/certificates/digital-certificate-revocation-35292.

[28] Y. Wang, D. Ruan, Dawu Gu, Jason Gao, Daming Liu, J. Xu, Fang Chen, Fei Dai, and Jinshi Yang. Analysis of smart grid security standards. In *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, volume 4, pages 697–701, June 2011.

[29] J. G. Wright and S. Wolthusen. Access Control and Availability Vulnerabilities in the ISO/IEC 61850 Substation Automation Protocol. In *Critical Information Infrastructures Security*, 2016.

[30] F. Zhao, Y. Hanatani, Y. Komano, B. Smyth, S. Ito, and T. Kambayashi. Secure authenticated key exchange with revocation for smart grid. In *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, pages 1–8, Jan 2012.