Issue of the Dangerous Delegated OCSP Responder Certificate

Kudelski Security - Nagravision SA – Kudelski Group – Nov. 2020

1. Introduction

In this memo, we address the issue of the "Dangerous Delegated OCSP Responder Certificate" [1]. We propose a new solution, different from the original solution suggested in the Mozilla Dev Security Policy (mdsp) mailing list.

This proposed solution addresses compliance issues, security risks, feasibility, time to implement and practicability (e.g. operational aspects) as well as negative financial consequences. For the feasibility, please consider the Proof of Concept provided in [2].

This memo is structured as follows. First, section 2 below describes and compares the two solutions. Then, section 3 analyzes them by providing an extensive list of potential concerns and discussing each of them in detail.

To illustrate the analysis further, a complete set of diagrams is available at [3].

2. Description of the initial situation and the two solutions

2.1. Initial situation

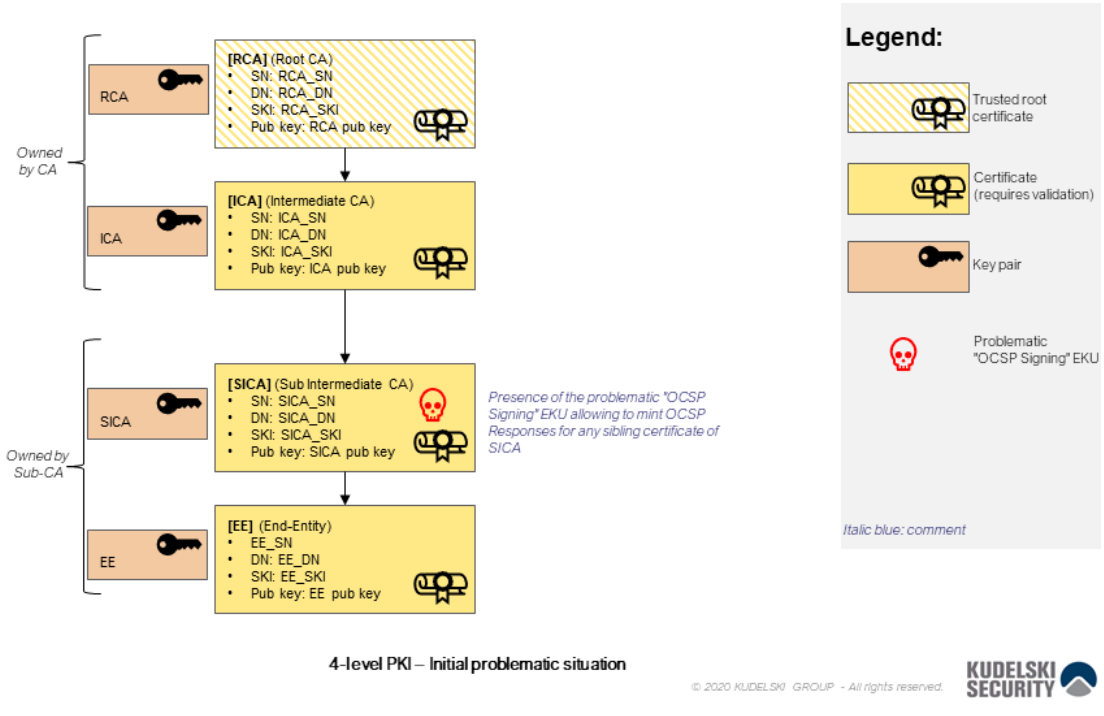

Figure 1: Initial situation - 4-level hierarchy

In Figure 1, SICA contains the problematic id-kp-ocspSigning Extended Key Usage (EKU). The goal is to reach a situation where the EE certificate can be verified up to the root CA in a chain where this EKU

extension is not present anymore. Indeed, the mere presence of this EKU makes SICA a delegated OCSP responder on behalf of ICA. If SICA was intended to be an issuing CA and not an OCSP responder, there is the security risk that SICA signs an OCSP response for any sibling certificate on behalf of ICA. This is a huge security concern if siblings are not operated by the same company/entity. This risk can impact all the Sub-CA companies having certificates under ICA. Hence, it is important that all the direct children certificates of ICA that have this problem are neutralized.

In addition to the security risk, there is also a compliance issue that is introduced because the Baseline Requirements [4] state that OCSP responders must also have the id-pkix-ocsp-nocheck extension in addition to the EKU.

2.2. Original solution

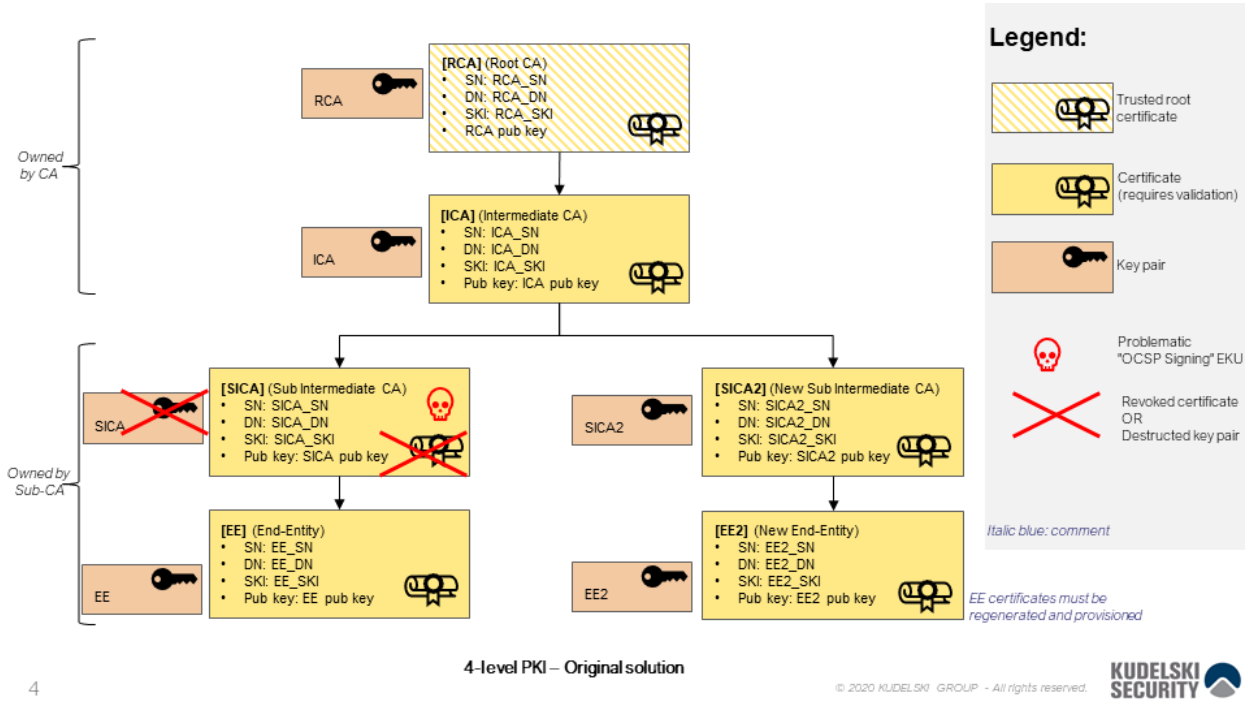

Figure 2: Original solution - 4-level hierarchy

The original solution addresses the issue in the following way (see Figure 2):

(i) a new SICA2 certificate is issued. It is signed by ICA and is based on a new key pair;

(ii) the old SICA certificate is revoked and, very importantly for security reasons, its associated private key is destroyed during an audited ceremony;

(iii) new EE2 certificates are issued by SICA2.

2.3. New solution

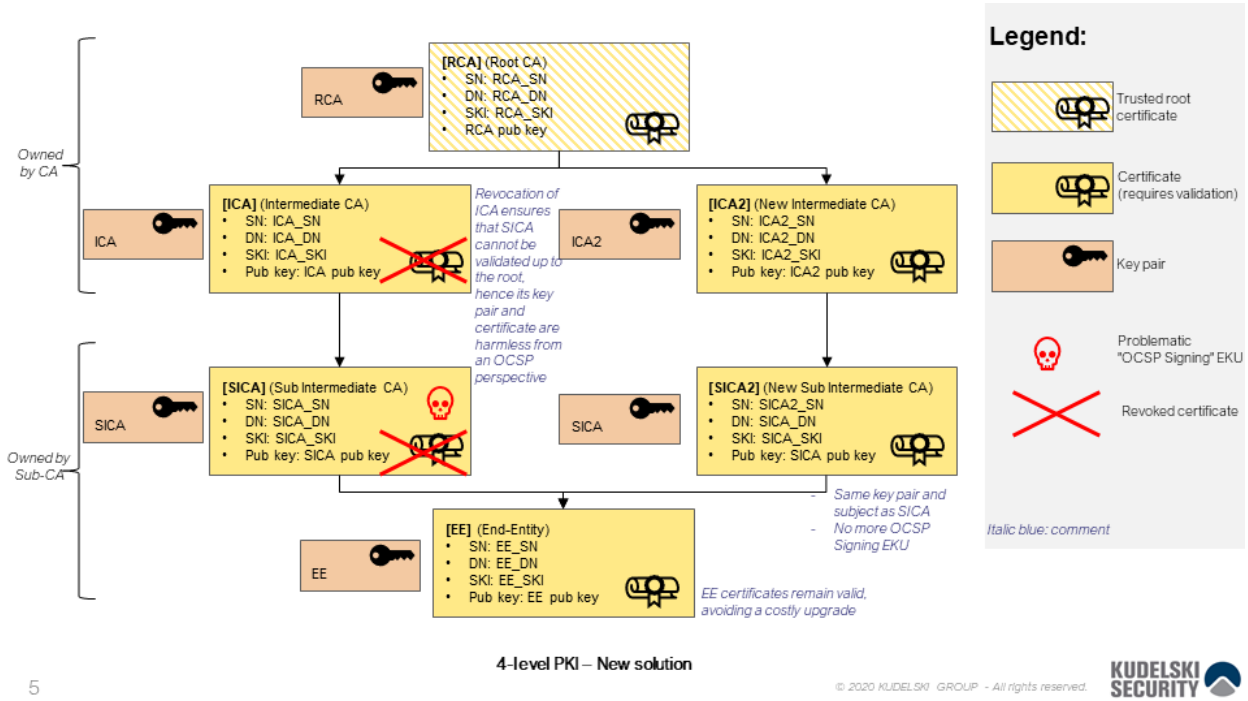

Figure 3: New solution - 4-level hierarchy

The new solution addresses the issue as illustrated in Figure 3, which can be summarized as follows:

(i) RCA issues a new ICA2 certificate, using a new DN and a new key pair;

(ii) RCA revokes the old ICA;

(iii) ICA2 issues a new SICA2 certificate, which reuses the same SICA DN and key pair but does not contain the OCSP problematic EKU;

(iv) SICA is also revoked.

Since the same DN and key pair are reused for SICA2, the EE certificate is still valid.

Also, the new SICA2 cannot be involved in any OCSP response, in any context at all (removal of the problematic EKU). The old SICA cannot be involved in any OCSP response either, in any context at all (it does not validate with regard to the new ICA2 and ICA is revoked). Finally, no third party, which would not have properly done the job of properly renewing a problematic SICA certificate, can do any harm to any other company (the old validation branch is not available anymore by the revocation of ICA).

2.4. Differences between the 2 solutions

2.4.1. 3-level vs 4-level hierarchy

The original solution works in both a 3-level hierarchy and 4-level hierarchy. The new solution works only in a 4-level (or more) hierarchy because that hierarchy allows for the parent of the affected certificate to be revoked because it is not a trust anchor. Therefore, no third party can be affected because nobody

trusts that chain anymore. This is impossible to achieve in the 3-level hierarchy because the parent of the affected certificate is a trust anchor and cannot be revoked.

2.4.2. Risk surface

However, each solution induces a different risk surface.

On the one hand, the new solution involves only one entity/company, the CA that issued the affected certificate. That CA must properly revoke its ICA certificate and regenerate a new one. The only risk is that they do not properly revoke the certificate that issued affected certificates.

On the other hand, the original solution requires destruction of private keys during audited ceremonies by all Sub-CAs concerned by the issue, independently. This means that a single Sub-CA failing to do so puts all the other Sub-CAs at risk. The resulting risk is the sum of the risks of each independent Sub-CA to fail to properly destroy its private key. That risk quickly increases as the number of independent Sub-CAs grows. Moreover, as the risk surface of a Sub-CA is usually larger than that of a CA since dealing with PKI is not its core activity but a technology it exploits, the situation is worsen.

Because of this exposure, the new solution has a smaller risk surface than the original solution.

2.4.3. Simplicity of the solution

The new solution is implemented using standard PKI procedures such as CRLs for revocation. A CRL is a simple and effective method to revoke a certificate and is industry battle-tested.

On the other hand, the original solution relies on key destruction ceremonies, which are complicated to implement and cannot ultimately prove that the risk is zero.

2.4.4. Time to implement

Since the original solution relies on key destruction ceremonies, it requires heavyweight coordination. Additionally, the number of end-entity certificates that need to be revoked can be huge for some CAs. For this reason, the time to implement the solution can be significantly longer.

The new solution can be implemented at a higher level in the hierarchy and does not require key destruction ceremonies. It requires revoking only a few certificates using simple CRLs. On top of that, only a few certificates need to be regenerated, thus considerably reducing the time required to implement compared to the original solution.

3. Eliminating security risks and compliance issues

Several concerns have been raised in the Mozilla Dev Security Policy [1]. The concerns were two-fold:

- A **compliance issue** (violation of a Baseline requirement).
- A **security risk**, allowing a delegated OCSP responder to sign responses for any sibling certificate, even if those siblings are not operated by the same entity.

In this section, we go through the raised concerns and address them with respect to the new solution.

3.1. Compliance issue

There is a violation of section 4.9.9 of the Baseline Requirements [4].

The proposed solution consists in revoking the affected SICA certificates and issue new ones without the id-kp-OCSPSigning EKU, which solves the compliance issue.

3.2. SICA siblings cannot be sure that the affected certificate is not going to unrevoke itself

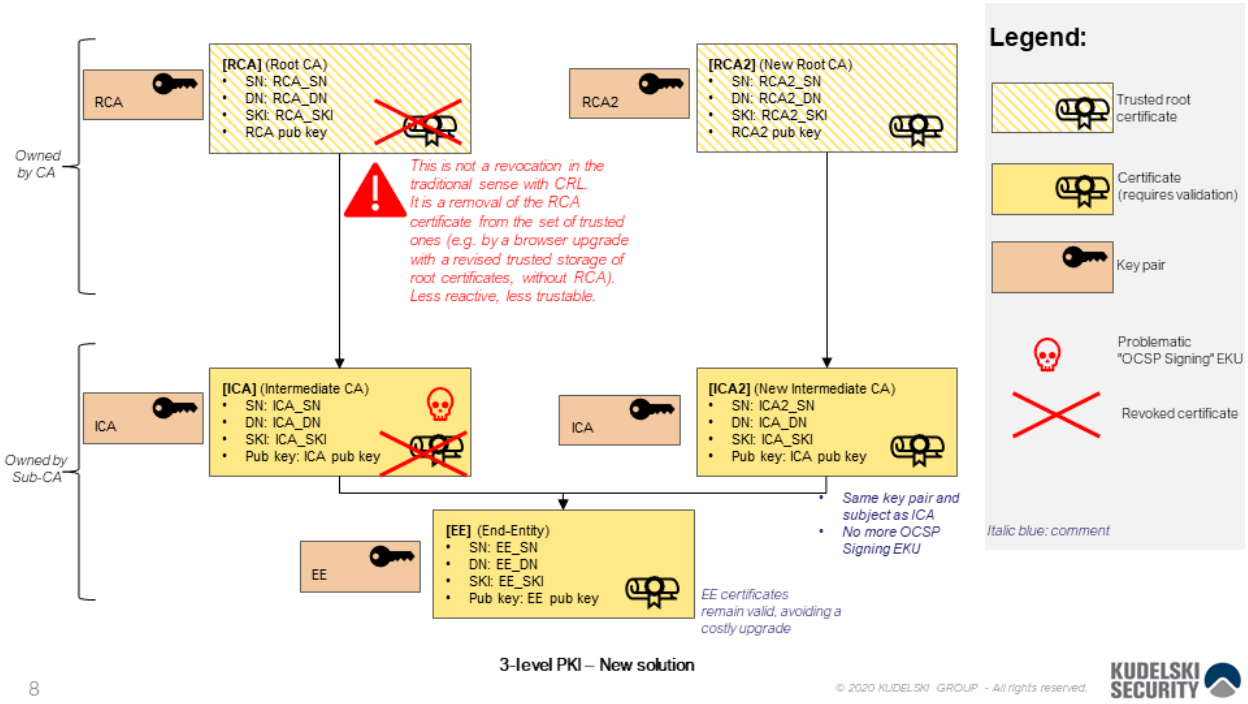

Figure 4: 3-level hierarchy issue

Assuming a 3-level certificate hierarchy such as the one in Figure 4 (RCA -> ICA -> EE), the concern is the following: if a new ICA2 is issued with the same key pair as ICA, then the private key was not destroyed and still exists. Therefore, it can still be used to sign a valid OCSP response for ICA itself or for any sibling in the certificate hierarchy. Thus, allowing ICA to unrevoke itself. That is also true even if a new RCA2 is created and used to sign the new ICA2 because that would still make ICA valid because RCA is a trust anchor and the only way to “revoke” it is to remove it from trust stores.

Contrarily to the 3-level hierarchy were the above concern applies, some affected CAs have a 4-level hierarchy such as the one shown in Figure 3 (RCA -> ICA -> SICA -> EE), where the affected certificate with the id-kp-OCSPSigning EKU is named SICA.

Since this is a 4-level hierarchy, the RCA in the 3-level hierarchy is now our ICA in this 4-level-hierarchy. Therefore, it is now possible to revoke the parent of the affected certificate SICA (its parent is ICA). The resulting operation does not require removing a trust anchor, which may take decades until everyone updates their systems, but truly mitigates the security risks immediately. Indeed, even if a malicious person used SICA2’s private key to sign an OCSP response to unrevoke SICA, since its parent (ICA) is also revoked and is not a trust anchor, the attack would not work.

3.3. Private keys may not have been properly destroyed

An audited ceremony is not an absolute proof that no other copy of the private key exists.

The community would benefit from revoking the parent (ICA) since some non-TLS issuing CAs may be unaudited and may have no other way to mitigate the security risk.

3.4. Offloading the pain to the community

Anything other than a key destruction can be perceived as unfair since it may be asking everyone (web browsers, applications, etc.) to change and adapt because someone failed to follow the expectations. It would put more work on the community to fix the problem by modifying how applications perform certificate validation.

However, the new solution only requires changes to a small subset of the hierarchy and does not offload the burden to the community. The new solution uses traditional PKI features such as CRLs and does not require the community to make any changes to certification validation checks.

3.5. Feasibility – Proof of Concept

We provide a proof-of-concept script written in Python that generates the hierarchy in Figure 3 and test that the unmodified EE certificate is still considered valid under the modified hierarchy using OpenSSL. The code and documentation are available on GitHub [2].

3.6. Practicability, adherence, and workload for the parties

The original solution requires that all Sub-CAs with problematic certificates at the SICA level engage in a very costly re-securization campaign. Non-concerned Sub-CAs have no work to do. In terms of security, it is important to stress that all Sub-CAs (the ones with problematic certificates and the ones without) remain at risk if a single Sub-CA does not do the proper work, or does it badly in terms of security.

The new solution requires all Sub-CAs to act, but at a reduced cost, since they must roll only intermediate certificates, which are quite few usually. It has impacts and cost, but clearly lower. And there is no risk for any Sub-CA at all, as the re-securization is managed at the ICA level by the CA.

4. Overall conclusion

In this document, a new solution was proposed and compared to the original solution. The feasibility of the new solution was verified by a proof-of-concept that people can use to test for themselves. Both the compliance issue and security risks were analyzed, without leading to any negative impacts while also reducing financial strain for some organizations. Depending on the exact and detailed nature of the hierarchy, the new solution has proven to bring operational advantages and is equivalent or improved with regard to security. We propose the community validate this new solution as well as our conclusions.

This analysis reflects the authors knowledge at the time of writing. In case concerns would be missing, the authors are willing to investigate them and complete the analysis. For that purpose, members of the community are encouraged to provide feedback in any form.

References

[1] "SECURITY RELEVANT FOR CAs: The curious case of the Dangerous Delegated Responder Cert", post by Ryan Sleevi on the Mozilla Dev Security Policy mailing list, https://groups.google.com/g/mozilla.dev.security.policy/c/EzjIkNGfVEE/m/XSfw4tZPBwAJ.

[2] Technical feasibility proof of concept, written in Python. Verification is performed using OpenSSL. https://github.com/kudelskisecurity/dangerous-ocsp-delegated-responder-cert-poc

[3] Detailed set of diagrams, file "OCSP_Signing_Issue_Figures.pdf". Available here: https://github.com/kudelskisecurity/dangerous-ocsp-delegated-responder-cert-poc/blob/master/doc/OCSP_Signing_Issue_Figures.pdf

[4] CA/Browser Forum, "Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates", version 1.7.1, August 20th 2020.

[5] RFC 6960, Internet Engineering Task Force (IETF), "X.509 Internet Public Key Infrastructure, Online Certificate Status Protocol – OCSP", available at https://tools.ietf.org/html/rfc6960.