Framing Questions - XML Encryption

Background information

Issue: There is an Oracle attack on AES in CBC mode that makes it practical to break the encryption. In some cases as few as ~100 calls (with manipulated plaintext). Use AE/AD ciphers, which combine integrity protection with data protection. Can manually CBC with MAC (this is what JOSE did, but not XML encryption). Requires AES-GCM which includes a MAC, which is the only practical (modern) bulk encryption technique safe for XML.

1. From the logout framing questions: Should logout requests/responses be required to be signed? If the answer is yes, then the SP needs to handle keys. This XML encryption discussion may drive the answer to this question.
2. Do we want to require encryption if we don't also require non-practically-breakable algorithms? (This could be an issue/limitation in 800-63, where XML encryption is required but not secure algorithms). May need other requirements listed as well.
3. In practice, confidentiality of user data in transit is provided by TLS or XML encryption (or both).
    a. Used in conjunction with a back channel protocol (such as artifact resolution or attribute query), TLS provides end-to-end confidentiality of user data in transit. In practice, XML encryption is not used on the back channel.
    b. The primary use of XML encryption today is in conjunction with SAML2 Web Browser SSO on a front channel. In that case, XML encryption is necessary for end-to-end confidentiality of user data in transit. OTOH, if you trust the browser (or don't care), TLS is sufficient (and of course TLS protects against other threats as well).
    c. An emerging use of XML encryption is in conjunction with SAML2 Single Logout but this is a completely new use case with little or no deployed base. (As Shibboleth IdP V3 proliferates, this may change.) We'll come back to this use case.
4. An important open question is: What protocols will the deployment profile mandate? What protocols will be discouraged (if any)?
    a. The use of XML encryption in conjunction with SSO is reasonably well understood. For example, if the deployment profile mandates support for SAML2 Artifact Resolution, the need to support inbound XML encryption at the SP is reduced. OTOH, if the deployment profile discourages the use of back-channel protocols (or ignores them altogether), the need to support inbound XML encryption at the SP is heightened.
    b. SLO is a more complicated use case. If the deployment profile mandates support for unconstrained SAML2 Single Logout, inbound XML encryption at the IdP is necessary to preserve privacy both in transit and at rest. The need to support inbound XML encryption at the IdP may be reduced by constraining the name identifier format and/or the binding used to transmit the LogoutRequest.
    c. Claim: The cost-benefit ratio associated with inbound XML encryption at the IdP is inordinately high. Can it be avoided?
5. A deployment that does not support seamless encryption key rollover does not fully support inbound XML encryption.
    a. Seamless encryption key rollover requires your SAML software to be configurable with multiple decryption keys. (Multiple encryption certificates in metadata do not facilitate encryption key rollover.)
    b. If your SAML software can not be configured with multiple decryption keys, and you publish an encryption certificate in metadata, seamless encryption key rollover is not possible. In that case, it is better not to publish an encryption certificate in metadata in the first place.
6. Possibly relevant facts about the Federation Manager:
    a. The FM requires an encryption certificate in SP metadata.
    b. The FM does not support an encryption certificate in IdP metadata.
    c. The FM does not require protocol endpoints in SP metadata to be HTTPS-protected.
    d. The FM requires protocol endpoints in IdP metadata to be HTTPS-protected.

Options:

1. XML encryption MUST NOT be used
    a. Arguments in favor
        i. Simplifies operational practices because SPs do not need keys (no key rollover either)
            1. Signed Authn requests can DoS an IDP by consuming compute resources
            2. Signing requests rarely adds value
        ii. Insecure algorithm (AES-CBC) doesn't provide confidentiality anyways
        iii. TLS is sufficient confidentiality
        iv. Simplifies the Deployment Profile by removing language around encryption
        v. Most attributes in a typical SAML assertion are not "private", though they are PII
    b. Arguments against
        i. SPs that currently require encryption may need manual reconfiguration
            1. Few SPs tend to require encryption, most will munch on the cleartext if they get it
2. XML encryption MUST be used and MUST use AES-GCM
    a. Arguments in favor
        i. Improved enforcement of audience restriction / bearer semantics because only intended SP can decrypt assertion
        ii. Increased confidentiality of attributes
        iii. Organization's security policy may require encryption of even partially sensitive data
        iv. NIST 800-63-3 requires encryption for Federation Assurance Level 2 or higher
        v. Profile compliance may be a lever to increase use of better algorithms
    b. Arguments against
        i. Some vendors/implementations can't meet this requirement, particularly if AES-CBC is disallowed
        ii. A wholesale migration to AES-GCM across the federation will be very costly
            1. On the other hand, the profile doesn't require that everybody in the federation support it
3. SPs MUST support either XML encryption or plaintext (the current situation) except that AES-CBC is NOT allowed
    a. Arguments in favor
        i. Gives SPs that can't handle AES-GCM an escape clause
        ii. Allows higher expectations for key handling and operational maturity for SPs if the rest can skip it
    b. Arguments against
        i. Forks profile with more complicated interoperability
        ii. Likely requires federation signaling of whether encryption should be used (can't count on IdPs supporting Shibboleth's "optional encryption" feature)
        iii. Leaves IdPs at the whim of the SPs in individual cases for their data protection profile