الاتحـــــــــــاد الدولـــــــــــي للاتصـــــــــــالات

ITU-T X.1080.0

(2017/03)

قطاع تقييس الاتصالات
في الاتحاد الدولي للاتصالات

السلسلة X: شبكات البيانات والاتصالات، بين الأنظمة المفتوحة ومسائل الأمن

أمن المعلومات والشبكات – القياس الحيوي عن بُعد

التحكم في النفاذ لحماية بيانات القياس الحيوي عن بُعد

التوصية ITU-T X.1080.0

الاتحاد الدولي للاتصالات

توصيات السلسلة X الصادرة عن قطاع تقييس الاتصالات

شبكات البيانات والاتصالات، بين الأنظمة المفتوحة ومسائل الأمن

الشبكات العمومية للبيانات	X.199–X.1
التوصيل البيني للأنظمة المفتوحة	X.299–X.200
التشغيل البيني للشبكات	X.399–X.300
أنظمة معالجة الرسائل	X.499–X.400
الدليل	X.599–X.500
التشغيل البيني لأنظمة التوصيل OSI ومظاهر النظام	X.699–X.600
إدارة التوصيل البيني للأنظمة المفتوحة (OSI)	X.799–X.700
الأمن	X.849–X.800
تطبيقات التوصيل البيني للأنظمة المفتوحة (OSI)	X.899–X.850
المعالجة الموزعة المفتوحة	X.999–X.900
أمن المعلومات والشبكات	
الجوانب العامة للأمن	X.1029–X.1000
أمن الشبكة	X.1049–X.1030
إدارة الأمن	X.1069–X.1050
الخصائص البيومترية	**X.1099–X.1080**
تطبيقات وخدمات آمنة	
أمن البث المتعدد	X.1109–X.1100
أمن الشبكة المحلية	X.1119–X.1110
أمن الخدمات المتنقلة	X.1139–X.1120
أمن الويب	X.1149–X.1140
بروتوكولات الأمن	X.1159–X.1150
الأمن بين جهتين نظيرتين	X.1169–X.1160
أمن معرفات الهوية عبر الشبكات	X.1179–X.1170
أمن التلفزيون القائم على بروتوكول الإنترنت	X.1199–X.1180
أمن الفضاء السيبراني	
الأمن السيبراني	X.1229–X.1200
مكافحة الرسائل الاقتحامية	X.1249–X.1230
إدارة الهوية	X.1279–X.1250
تطبيقات وخدمات آمنة	
اتصالات الطوارئ	X.1309-X.1300
أمن شبكات المحاسيس واسعة الانتشار	X.1339–X.1310
التوصيات المتعلقة بالبنية التحتية للمفاتيح العمومية	X.1349–X.1340
أمن إنترنت الأشياء (IoT)	X.1369–X.1360
أمن أنظمة النقل الذكية (ITS)	X.1379–X.1370

تبادل معلومات الأمن السيبراني	
نظرة عامة على الأمن السيبراني	X.1519–X.1500
تبادل مواطن الضعف/الحالة	X.1539–X.1520
تبادل الأحداث/الأحداث العارضة/المعلومات الحدسية	X.1549–X.1540
تبادل السياسات	X.1559–X.1550
طلب المعلومات الحدسية والمعلومات الأخرى	X.1569–X.1560
تعرف الهوية والاكتشاف	X.1579–X.1570
التبادل المضمون	X.1589–X.1580
أمن الحوسبة السحابية	
نظرة عامة على أمن الحوسبة السحابية	X.1601–X.1600
تصميم أمن الحوسبة السحابية	X.1639–X.1602
أفضل الممارسات ومبادئ توجيهية بشأن أمن الحوسبة السحابية	X.1659–X.1640
تنفيذ أمن الحوسبة السحابية	X.1679–X.1660
أمن أشكال أخرى للحوسبة السحابية	X.1699–X.1680

لمزيد من التفاصيل يرجى الرجوع إلى قائمة التوصيات الصادرة عن قطاع تقييس الاتصالات.

التوصية ITU-T X.1080.0

التحكم في النفاذ لحماية بيانات القياس الحيوي عن بُعد

ملخص

تقدم التوصية ITU-T X.1080.0 مواصفات بشأن كيفية حماية معلومات القياس الحيوي عن بُعد من النفاذ غير المصرح به. وهي تفعل ذلك باتخاذ منظور خدمي، حيث لا تقدَّم إلا المعلومات اللازمة لغرض معين، أي لا يتاح الاطلاع على أساس *حق المعرفة* فحسب، بل أيضاً على أساس *الحاجة إلى المعرفة*.

ويرد في الصميم من هذه التوصية توصيف نعت مدرج في شهادة نعت أو شهادة المفتاح العمومي التي توصِّف بالتفصيل ماهية الامتيازات التي يملكها كيان معين واحد أو أكثر من أنماط الخدمة.

ويوفَّر الأمن باستخدام ملف تعريف قواعد تركيب الرسالة التجفيرية (CMS). ويوفَّر ملف تعريف CMS الاستيقان والسلامة، وعند الاقتضاء، الكتمان (التجفير).

ويهدف ملف التعريف هذا إلى تقديم الدعم الأمني لمواصفات بيانات القياس الحيوي عن بُعد بشكل عام. ويفترض ملف التعريف النشر الصحيح للبنية التحتية للمفتاح العمومي (PKI)، ويعتمد على ذلك.

وتعتمد هذه التوصية أيضاً على نشر البنية التحتية لإدارة الامتياز (PMI).

التسلسل التاريخي

الطبعة	التوصية	تاريخ الموافقة	لجنة الدراسات	معرف الهوية الفريد*
1.0	ITU-T X.1080.0	2017-03-30	17	11.1002/1000/13193

مصطلحات أساسية

التحكم في النفاذ، ديفي–هيلمان (Diffie-Hellman)، البنية التحتية للمفتاح العمومي، بيانات القياس الحيوي عن بُعد.

* للنفاذ إلى توصية، ترجى كتابة العنوان http://handle.itu.int/ في حقل العنوان في متصفح الويب لديكم، متبوعاً بمعرف التوصية الفريد. ومثال ذلك، http://handle.itu.int/11.1002/1000/11830-en.

تمهيد

الاتحاد الدولي للاتصالات وكالة متخصصة للأمم المتحدة في ميدان الاتصالات وتكنولوجيات المعلومات والاتصالات (ICT). وقطاع تقييس الاتصالات (ITU-T) هو هيئة دائمة في الاتحاد الدولي للاتصالات. وهو مسؤول عن دراسة المسائل التقنية والمسائل المتعلقة بالتشغيل والتعريفة، وإصدار التوصيات بشأنها بغرض تقييس الاتصالات على الصعيد العالمي.

وتحدد الجمعية العالمية لتقييس الاتصالات (WTSA) التي تجتمع مرة كل أربع سنوات المواضيع التي يجب أن تدرسها لجان الدراسات التابعة لقطاع تقييس الاتصالات وأن تُصدر توصيات بشأنها.

وتتم الموافقة على هذه التوصيات وفقاً للإجراء الموضح في القرار 1 الصادر عن الجمعية العالمية لتقييس الاتصالات.

وفي بعض مجالات تكنولوجيا المعلومات التي تقع ضمن اختصاص قطاع تقييس الاتصالات، تُعد المعايير اللازمة على أساس التعاون مع المنظمة الدولية للتوحيد القياسي (ISO) واللجنة الكهرتقنية الدولية (IEC).

ملاحظة

تستخدم كلمة "الإدارة" في هذه التوصية لتدل بصورة موجزة سواء على إدارة اتصالات أو على وكالة تشغيل معترف بها.

والتقيد بهذه التوصية اختياري. غير أنها قد تضم بعض الأحكام الإلزامية (بهدف تأمين قابلية التشغيل البيني والتطبيق مثلاً). ويعتبر التقيّد بهذه التوصية حاصلاً عندما يتم التقيّد بجميع هذه الأحكام الإلزامية. ويستخدم فعل "يجب" وصيغ ملزمة أخرى مثل فعل "ينبغي" وصيغها النافية للتعبير عن متطلبات معينة، ولا يعني استعمال هذه الصيغ أن التقيّد بهذه التوصية إلزامي.

حقوق الملكية الفكرية

يسترعي الاتحاد الانتباه إلى أن تطبيق هذه التوصية أو تنفيذها قد يستلزم استعمال حق من حقوق الملكية الفكرية. ولا يتخذ الاتحاد أي موقف من القرائن المتعلقة بحقوق الملكية الفكرية أو صلاحيتها أو نطاق تطبيقها سواء طالب بها عضو من أعضاء الاتحاد أو طرف آخر لا تشمله عملية إعداد التوصيات.

وعند الموافقة على هذه التوصية، لم يكن الاتحاد قد تلقى إخطاراً بملكية فكرية تحميها براءات الاختراع يمكن المطالبة بها لتنفيذ هذه التوصية. ومع ذلك، ونظراً إلى أن هذه المعلومات قد لا تكون هي الأحدث، يوصى المسؤولون عن تنفيذ هذه التوصية بالاطلاع على قاعدة البيانات الخاصة ببراءات الاختراع في مكتب تقييس الاتصالات (TSB) في الموقع http://www.itu.int/ITU-T/ipr/.

جدول المحتويات

الصفحة

مقدمة

ينطوي جمع بيانات القياس الحيوي عن بُعد من الأفراد على خطر انتهاك الخصوصيات.

وقد تتعدد أسباب حماية هذه المعلومات. فقد تتيح المعلومات النفاذ إلى شركة أو منظمة أو قد تكون ذات طبيعة حساسة تقيد توزيعها.

وللحماية من الكشف غير المرغوب فيه لبيانات القياس الحيوي عن بُعد جانبان رئيسيان:

- حماية البيانات أثناء الإرسال، عن طريق التجفير عادة، وحماية البيانات المخزنة؛
- التحكم في النفاذ إلى البيانات المخزنة.

وفي حين ينبغي أن تكون أنظمة القياس الحيوي عن بُعد على مستوى عال من الأمن فيما يتعلق بالكتمان (التجفير)، والاستيقان، والسلامة، والحماية المادية، واستخدام جدران الحماية وبرامج الحماية من الفيروسات، وما إلى ذلك، تقتضي الضرورة أيضاً إنشاء نظام محكم للتحكم في النفاذ من أجل النفاذ إلى المعلومات المخزنة، ولا سيما المعلومات عن الأفراد. ويكتسي هذا الجانب الأخير أهمية خاصة في أنظمة القياس الحيوي عن بُعد.

وتعاني مخططات التحكم في النفاذ العامة من قصور يتمثل في أنها تنظر أساساً في قبول (أو رفض) حق الحصول على المعلومات، ولكنها لا تنظر في تفاصيل جانب *الحاجة إلى المعرفة*. وتعني الحاجة إلى المعرفة ضمناً عدم كفاية الحق في الاطلاع على البيانات، بل لا بد من إثبات أن المعلومات ستُستخدم لأغراض مشروعة.

وينبغي ألا تقدَّم المعلومات إلا للاستخدام المقصود منها. فتُجمع المعلومات الطبية عن مريض لإتاحة العلاج الأمثل لهذا المريض، وينبغي عدم استخدامها لأي غرض آخر، إلا ربما في المشاريع البحثية المضبوطة بإحكام التي قد تتطلب استخدام بعض المعلومات من مجموعة معينة من المرضى، وخلاف ذلك، تتعين حماية المعلومات من البحث الجارف عن المعلومات.

وهناك نوعان رئيسيان من التحكم في النفاذ: مادي ومنطقي. فيقيِّد التحكم في النفاذ المادي النفاذ إلى الجامعات والمباني والغرف وأصول تكنولوجيا المعلومات (IT) المادية. أما النفاذ المنطقي فيقيِّد التوصيلات بشبكات الحاسوب وملفات وبيانات النظام. ولا تنظر هذه التوصية إلا في التحكم في النفاذ المنطقي.

ويشمل التحكم في النفاذ استيقان مقدم الخدمة من النافذين بصورة مأمونة. وتفترض هذه التوصية استخدام التوقيعات الرقمية وبنية تحتية قائمة للمفتاح العمومي (PKI).

ويمكن لمواصفات بيانات القياس الحيوي عن بُعد الأخرى أن تستشهد مرجعياً بالتوصية ITU-T X.1080.0.

ويوصِّف الملحق A، الذي يشكل جزءاً أساسياً من هذه التوصية، توزيع معرفات الكائنات التي تستخدمها سلسلة التوصيات ITU-T X.1080.

ويقدم الملحق B، الذي يشكل جزءاً أساسياً من هذه التوصية، مواصفة بيانات القياس الحيوي عن بُعد في قواعد تركيب الرسالة التجفيرية (CMS)، كما نوقشت في المعيار IETF RFC 5652 المستخدم في هذه التوصية. ويمكن لمواصفات بيانات القياس الحيوي عن بُعد الأخرى أيضاً أن تستخدمه كمرجع.

ويقدم الملحق C، الذي يشكل جزءاً أساسياً من هذه التوصية، توصيفاً رسمياً لبروتوكولات تأكيد الامتياز والتخصيص في شكل وحدة قواعد التركيب المجردة رقم 1 (ASN.1).

ويقدم التذييل الأول، الذي لا يشكل جزءاً أساسياً من هذه التوصية، توصيفاً غير رسمي لملف تعريف قواعد تركيب الرسالة التجفيرية (CMS) في شكل وحدة قواعد التركيب المجردة رقم 1 (ASN.1).

التوصية ITU-T X.1080.0

التحكم في النفاذ لحماية بيانات القياس الحيوي عن بُعد

1 مجال التطبيق

تقدم هذه التوصية مواصفة بشأن الكيفية التي يمكن بها حماية الخصوصيات في بيئة بيانات القياس الحيوي عن بُعد باستخدام التحكم في النفاذ إلى بيانات القياس الحيوي عن بُعد استناداً إلى الخصوصيات (ACT). وإذ لا يمكن لهذه التوصية تحديد جميع أنماط المعلومات الممكنة، فإن مجال تطبيقها يشمل تقديم أدوات عامة للتعامل مع جميع أنماط المعلومات بطريقة آمنة. ويشمل ذلك تعريف بروتوكول لتخصيص الامتيازات وبروتوكول للنفاذ إلى المعلومات باستخدام تأكيد الامتياز. وتقدم هذه التوصية مبادئ توجيهية ولا تتضمن متطلبات الامتثال.

وتقع الجوانب التالية خارج مجال تطبيق هذه التوصية:

- الحماية المادية للمعلومات؛
- النفاذ غير المخوَّل لموظف التشغيل الذي يدير النظام الأمني، والذي يمكنه بالتالي الالتفاف على التدابير الأمنية.

2 المراجع

تتضمن التوصيات التالية لقطاع تقييس الاتصالات وغيرها من المراجع أحكاماً تشكل من خلال الإشارة إليها في هذا النص جزءاً أساسياً من هذه التوصية. وقد كانت جميع الطبعات المذكورة سارية الصلاحية في وقت النشر. ولما كانت جميع التوصيات والمراجع الأخرى تخضع إلى المراجعة، يرجى من جميع المستعملين لهذه التوصية السعي إلى تطبيق أحدث طبعة للتوصيات والمراجع الأخرى الواردة أدناه. وتُنشر بانتظام قائمة توصيات قطاع تقييس الاتصالات السارية الصلاحية. والإشارة إلى وثيقة ما في هذه التوصية لا يضفي على الوثيقة في حد ذاتها صفة التوصية.

[ITU-T X.500-series] Recommendation ITU-T X.5xx (2016) | ISO/IEC 9594-x series, *Information technology – Open Systems Interconnection – The Directory*.

[ITU-T X.501] Recommendation ITU-T X.501 (2016) | ISO/IEC 9594-2, *Information technology – Open Systems Interconnection – The Directory: Models*.

[ITU-T X.509] Recommendation ITU-T X.509 (2016) | ISO/IEC 9594-8, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks*.

[ITU-T X.520] Recommendation ITU-T X.520 (2016) | ISO/IEC 9594-6, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types*.

[ITU-T X.521] Recommendation ITU-T X.521 (2016) | ISO/IEC 9594-7, *Information technology – Open Systems Interconnection – The Directory: Selected object classes*.

[ITU-T X.680] Recommendation ITU-T X.680 (2015) | ISO/IEC 8824-1, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*.

[ITU-T X.681] Recommendation ITU-T X.681 (2015) | ISO/IEC 8824-2, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.

[ITU-T X.682] Recommendation ITU-T X.682 (2015) | ISO/IEC 8824-3, Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.

[ITU-T X.683] Recommendation ITU-T X.683 (2015) | ISO/IEC 8824-4, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications*.

[ITU-T X.690] Recommendation ITU-T X.690 (2015) | ISO/IEC 8825-1, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.

[ITU-T X.1080.1] Recommendation ITU-T X.1080.1 (2011), *e-Health and world-wide telemedicines – Generic telecommunication protocol*.

[ITU-T X.1081] Recommendation ITU-T X.1081 (2011), *The telebiometric multimodal model – A framework for the specification of security and safety aspects of telebiometrics*.

[IETF RFC 2631] IETF RFC 2631 (1999), *Diffie-Hellman Key Agreement Method*.

[IETF RFC 3185] IETF RFC 3185 (2001), *Reuse of CMS Content Encryption Keys*.

[IETF RFC 5083] IETF RFC 5083 (2007), *Cryptographic Message Syntax (CMS) - Authenticated-Enveloped-Data Content Type*.

[IETF RFC 5652] IETF RFC 5652 (2009), *Cryptographic Message Syntax (CMS)*.

[IETF RFC 5753] IETF RFC 5753 (2010), *Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)*.

[IETF RFC 5911] IETF RFC 5911 (2010), *New ASN.1 Modules for Cryptographic Message Syntax (CMS) and S/MIME*.

[IETF RFC 6268] IETF RFC 6268 (2011), *Additional New ASN.1 Modules for the Cryptographic Message Syntax (CMS) and the Public Key Infrastructure Using X.509 (PKIX)*.

3 التعاريف

1.3 المصطلحات المعرَّفة في وثائق أخرى

تستخدم هذه التوصية المصطلحات التالية المعرَّفة في وثائق أخرى:

1.1.3 شهادة نعت (attribute certificate) [ITU-T X.509]: بنية من البيانات تحمل التوقيع الرقمي لسلطة نعت، وتربط بعض قيم النعت بمعلومات عن تعرّف هوية حامل الشهادة.

2.1.3 نمط النعت (attribute type) [ITU-T X.501]: هو ذلك المكوّن من النعت الذي يبيّن صنف المعلومات التي يختزنها النعت.

3.1.3 قيمة النعت (attribute value) [ITU-T X.501]: هي حالة معيّنة من صنف المعلومات يبيّنها نمط النعت.

4.1.3 امتياز (privilege) [ITU-T X.509]: هو نعت أو صفة تسندها سلطة ما إلى كيان ما.

5.1.3 حامل الامتياز (privilege holder) [ITU-T X.509]: كيان خُصص بامتياز. ويمكن لحامل الامتياز أن يؤكد امتيازه لغرض معين.

6.1.3 متحقق من الامتياز (privilege verifier) [ITU-T X.509]: كيان يتحقق من الشهادات وفق سياسة الامتياز.

7.1.3 مَصْدر السلطة (SOA) [ITU-T X.509]: سلطة نعت يستطيع متحقق من الامتياز أن يمنحها الثقة بشأن مورد معين، باعتبارها السلطة النهائية التي تخصص مجموعة من الامتيازات لتأكيد ذلك المورد.

2.3 تعاريف معرّفة في هذه التوصية

تعرّف هذه التوصية المصطلحات التالية:

1.2.3 التحكم في النفاذ (access control): تقنية أمنية تستخدم لتنظيم مَن يستطيع فعل ماذا في مصادر المعلومات في بيئة حوسبة.

2.2.3 خدمة نفاذ (access service): خدمة يقدمها مقدم الخدمة لتنفيذ معاملة معينة.

3.2.3 نافذ (accessor): حامل امتياز يَنفذ إلى خدمة نفاذ معينة باستخدام امتيازه.

4.2.3 نعت (attribute): معلومة من نمط معين ترتبط بكائن. وتتكون المعلومات المقترنة بكائن من نعوت.

5.2.3 ميدان حماية البيانات (data protection domain): ميدان تكون فيه المعلومات محمية تحت مكون إدارة واحد.

6.2.3 الاسم المميز (distinguished name): اسم يحدد هوية كائن فريد ضمن سياق محدد ويتكون من واحد أو أكثر من مكونات اسم تعبر عن موضع الكائن في تراتبية كائنات.

7.2.3 كائن (object): شخص أو دائرة أو مهني أو أي نمط آخر من الكائنات توجد معلومات عنه ويمكن التعرف عليه باسم مميز.

8.2.3 صنف الكائن (object class): عائلة كائنات معرّفة الهوية تشترك بخصائص معيّنة

9.2.3 عملية (operation): تفاعل يتألف من طلب ورد بين نافذ ومقدم الخدمة لهدف معين.

10.2.3 مواصفة (specification): توصية من قطاع تقييس الاتصالات أو معيار دولي أو أي مواصفة وضعتها منظمة معترف بها لوضع المعايير.

4 المختصرات والأسماء المختصرة

تستخدم هذه التوصية المختصرات والأسماء المختصرة التالية:

AA سلطة نعت *(Attribute Authority)*

ABAC التحكم في النفاذ القائم على النعت *(Attribute-based Access Control)*

ACL قائمة التحكم في النفاذ *(Access Control List)*

ACT قائمة التحكم في النفاذ لبيانات القياس الحيوي عن بُعد *(Access Control for Telebiometrics)*

AES معيار التجفير المتقدم *(Advanced Encryption Standard)*

ASN.1 قواعد التركيب المجردة رقم 1 *(Abstract Syntax Notation One)*

BER قواعد التشفير الأساسية *(Basic Encoding Rules)*

CA سلطة إصدار الشهادات *(Certification Authority)*

CEK مفتح تجفير المحتوى *(Content Encryption Key)*

CMS قواعد تركيب الرسالة التجفيرية *(Cryptographic Message Syntax)*

DER قواعد التشفير المميزة *(Distinguished Encoding Rules)*

DH ديفي–هيلمان *(Diffie-Hellman)*

ECC تجفير المنحني الإهليلجي *(Elliptic Curve Cryptography)*

ECDH ديفي–هيلمان المنحني الإهليلجي *(Elliptic Curve Diffie-Hellman)*

GCM أسلوب غالوا (Galois)/العداد *(Galois/Counter Mode)*

KEK مفتاح تجفير المفتاح *(Key-Encryption Key)*

LDAP البروتوكول الخفيف للنفاذ إلى الدليل *(Lightweight Directory Access Protocol)*

MAC شفرة الاستيقان من الرسالة *(Message Authentication Code)*

PDU وحدة بيانات البروتوكول *(Protocol Data Unit)*

PKI البنية التحتية للمفتاح العمومي *(Public-Key Infrastructure)*

PMI البنية التحتية لإدارة الامتياز *(Privilege Management Infrastructure)*

SDO منظمة وضع معايير *(Standards Developing Organization)*

SOA مصدر السلطة *(Source of Authority)*

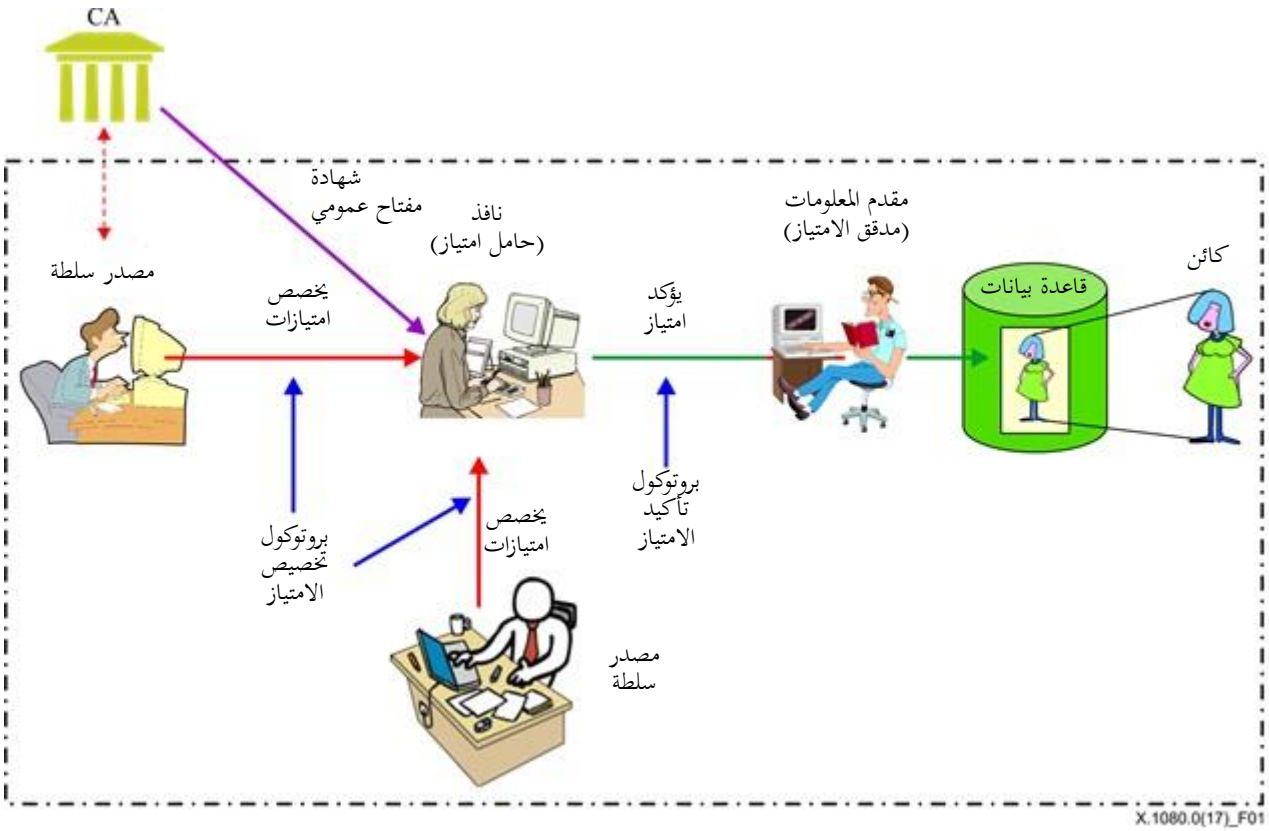
5 الاصطلاحات

تعرض هذه التوصية ترميز ASN.1 بالنمط الداكن من الحرف الطباعي **courier new typeface**، فإن وردت أنماط وقيم ترميز ASN.1 في متن النص العادي فهي تتميز عن النص العادي بعرضها بالنمط الداكن من الحرف الطباعي Courier New Typeface.

وإن كانت البنود المدرجة في قائمة مرقّمة (بدلاً من استعمال "–" أو أحرف) فسوف تُعتبر البنود خطوات ضمن إجراء.

6 مفاهيم ونماذج أساسية

1.6 الحماية في ميدان حماية بيانات واحد



الشكل 1 – نموذج ميدان حماية بيانات واحد

يوضح الشكل 1 مختلف الشركاء والبروتوكولات في بيئة تحكم في النفاذ ضمن ميدان حماية بيانات واحد. ويتكون ميدان حماية البيانات من جميع الكيانات المشاركة في الحماية والمندرجة تحت إدارة مشتركة.

ويمكن لكيان يدعى النافذ أن يتولى وظيفة تتطلب الإذن لتنفيذ عمليات على معلومات عن كائنات يختزنها مقدم المعلومات. وكمثال على ذلك، يحتاج طبيب مسؤول عن مريض للنفاذ إلى السجل الصحي للمريض لإعطاء العلاج الاختياري، في حين لا يحتاج أطباء آخرون غير معنيين بالمريض لمثل هذه المعلومات.

وينبغي ألا يعطى الإذن لتنفيذ عملية معينة على المعلومات إلا إذا كان للنافذ *حق وحاجة حقيقية* للقيام بهذه العملية، أي إذا كان مخصصاً *بالامتياز* اللازم.

ولا يشمل مجال تطبيق هذه التوصية توصيف ماهية شروط منح الامتياز إلى الكيانات. وينحصر مجال التطبيق في تقديم الأدوات اللازمة لإدارة الامتيازات بطريقة آمنة.

ويحتاج ميدان حماية البيانات لإنشاء واحدة أو أكثر من السلطات لتخصيص امتيازات لكيانات. ويستخدم هنا مصطلح مصدر السلطة (SOA) المعرف في التوصية [ITU-T X.509] كمصطلح يعبر عن السلطة النهائية لتخصيص امتيازات في مجال محدد ضمن ميدان حماية البيانات.

وتستخدم هذه التوصية شهادات لتخصيص امتياز إلى نافذين. ويمكن حمل الامتياز في شهادات نعت ضمن مكون **نعوت** (**attributes**) أو في شهادات المفتاح العمومي ضمن توسعة **subjectDirectoryAttributes**. ويمكن عادة تقديم الامتيازات الدائمة في شهادات المفتاح العمومي، في حين يمكن عادة تقديم الامتيازات المؤقتة في شهادات نعت.

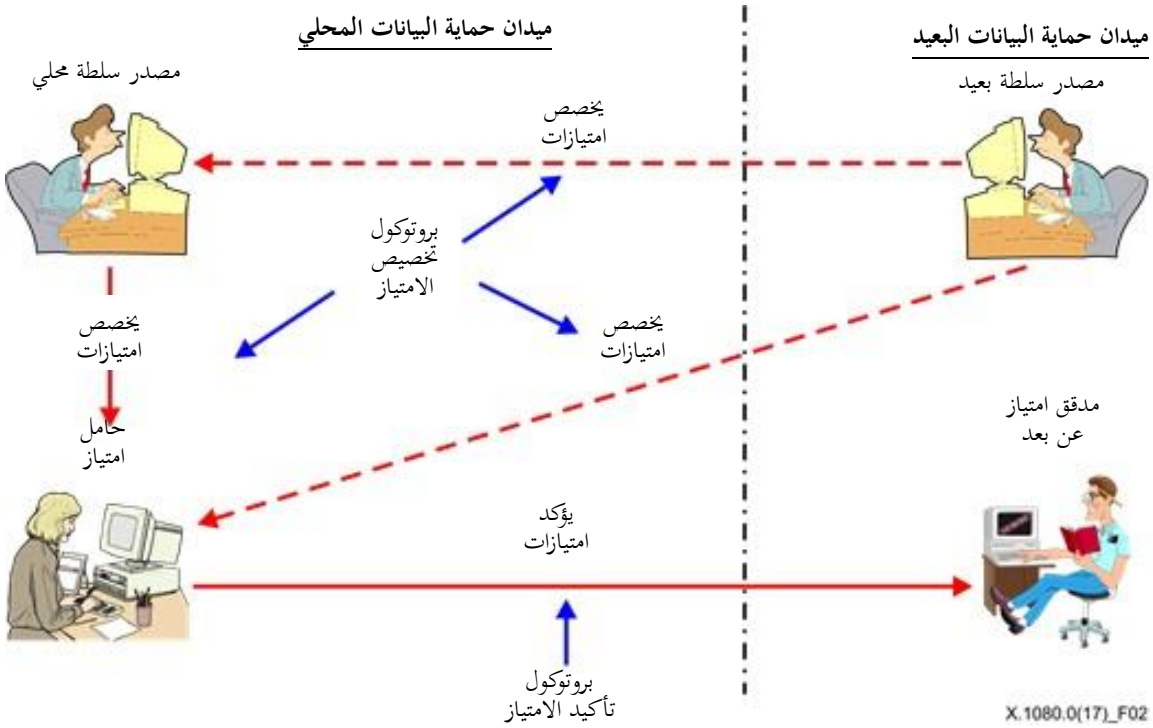
ويوضح الشكل 1 العلاقة بين المكونات المختلفة في ميدان حماية بيانات واحد. ويظهر مصدري سلطة يتولى كل منهما مسؤولية تخصيص امتيازات ذات جوانب مختلفة لحملة الامتياز.

وقد تلزم بعض الامتيازات للعمليات اليومية للنافذ، وبالتالي فهي ذات طبيعة أكثر دواماً، في حين تخصَص امتيازات أخرى لمعالجة الأوضاع الخاصة، وبالتالي فهي ذات طبيعة مؤقتة.

وعندما تخصص امتيازات لنافذ، يصبح حاملاً لامتياز ويمكن أن يستخدم امتيازه للنفاذ إلى معلومات من خلال بروتوكول تأكيد الامتياز. ويتحقق مدقق الامتيازات الممثل لمقدم المعلومات من الامتياز المدعى قبل السماح بالنفاذ إلى المعلومات.

ويمكن لمصدر السلطة (SOA) أن يسيِّر الامتيازات إما بوسيلة محلية أو عن طريق دمجها في شهادة نعت موقَّعة في بروتوكول تخصيص امتياز على النحو الموضح في الشكل 1.

2.6 ميدان حماية البيانات المتقاطعة



الشكل 2 – نموذج ميدان حماية البيانات المتقاطعة

يوضح الشكل 2 حالة يحتاج فيها نافذ للنفاذ إلى المعلومات في ميدان آخر لحماية البيانات، مثلاً، لعلاج مريض ترد المعلومات الصحية الحيوية التي تخصه في ميدان آخر لحماية البيانات.

وفي هذه البيئة، يطلب النافذ، إما مباشرة أو من خلال طلبات مصدر سلطة محلية، النفاذ إلى معلومات محددة بشأن كائن معين. ولا يشمل مجال تطبيق هذه التوصية كيف يوجَّه هذا الطلب، ولكن يمكن أن يكون عن طريق البريد الإلكتروني أو عن طريق الهاتف.

ويصدر مصدر سلطة بعيد شهادة نعت بنعت من نمط `accessService` يحمل الامتياز اللازم. ويوقِّع مصدر السلطة البعيد شهادة النعت هذه. وقد يكون **حامل** (`holder`) شهادة النعت إما مصدر سلطة محلي أو الكيان الذي يحتاج الامتياز.

فإذا كان مصدر السلطة المحلي هو صاحب الامتياز المخصص، فإن مصدر السلطة هذا يقوم بما يلي:

أ) فك تجفير المحتوى إذا طُلب ذلك؛

ب) التحقق من التوقيع على المحتوى؛

ج) استخراج شهادة النعت من المحتوى؛

د) إقرار صحة شهادة النعت المستخرجة؛

هـ) إنشاء شهادة نعت جديدة بمعلومات الامتياز المأخوذة من شهادة النعت الواردة، حيث حامل الامتياز هو الكيان المحتاج لهذا الامتياز؛

و) ثم تسيَّر شهادة النعت هذه إلى جانب شهادة النعت الواردة من مصدر السلطة البعيد إلى الكيان الذي يحتاج إلى الامتياز (أي حامل الامتياز).

3.6 النموذج خدمي المنحى

إن التحكم في النفاذ إلى بيانات القياس الحيوي عن بُعد هو تحكم خدمي المنحى بمعنى أن النافذ يمكنه أن يستحضر مجموعة من الوظائف المتميزة تدعى *خدمات النفاذ*. وخدمة النفاذ هي وظيفة يقدمها مقدم خدمة قادر على أداء خدمة نفاذ معينة. ولعل ثمة مثالاً على خدمة النفاذ في القدرة على التعامل مع المعلومات الخاصة بالمريض داخل مستشفى. ويُستخدم معرف الكائن لتحديد نمط خدمة نفاذ معينة. ولا تحدد هذه التوصية خدمات نفاذ بعينها، ولكنها توفر الأدوات اللازمة لإنشاء خدمات النفاذ. ويمكن لمواصفات الإحالة المرجعية أن تعرِّف خدمات النفاذ ذات الصلة بنطاقها.

وعلى الرغم من أن للنافذ حق النفاذ إلى خدمة معينة، فقد لا يملك الامتياز الذي يغطي جميع جوانب هذه الخدمة.

4.6 نموذج الكائن والنعت

صُمم نموذج المعلومات المحدد في التوصية [ITU-T X.501] للتعامل مع هياكل البيانات المختلفة. وتستخدم هذه التوصية نموذج المعلومات هذا للتحكم في النفاذ.

وفي نموذج ITU-T X.501، تنظَّم الكائنات المزمع حمايتها في أصناف كائنات، حيث تمتلك الكائنات ضمن صنف كائن خصائص مشتركة ذات صلة في سياق معين. ويحدَّد صنف الكائن بمعرف الكائن. وتتوسع التوصية [ITU-T 1080.1] في هذا المفهوم من خلال تخصيص معرفات الكائن لمجموعة من أصناف كائنات في فئات ذات صلة ببيانات القياس الحيوي عن بُعد.

وتعرِّف التوصية [ITU-T X.521] أيضاً مجموعة من أصناف كائنات عامة صالحة للاستعمال. وفي الحالات التي تطلّب صنف كائن لا علاقة مباشرة له ببيانات القياس الحيوي عن بُعد، ينبغي أن يُستخدم صنف كائن سبق تعريفه كلما أمكن ذلك.

وتعرَّف فرادى الكائنات بأسماء مميزة على النحو المحدد في التوصية [ITU-T X.501]. ويتكون الاسم المميز من واحد أو أكثر من مكونات الاسم الذي يعبر عن موضعه ضمن تراتبية الكائن.

وتنمذَج المعلومات المرتبطة بكائن كمجموعة من النعوت. وتشكل النعوت ذات الخصائص المشتركة نمط نعت. ويحدَّد نمط نعت بمعرف الكائن. والنعت هو تتابع معرف كائن يحدد نمط النعت وقيمة واحدة أو أكثر لهذا النمط. ولا يجوز أن يسنَد إلى كائن إلا نعت واحد من نمط معين. وتعرِّف التوصية [ITU-T X.520] مجموعة صالحة للاستعمال من أنماط النعت العامة. وينبغي أن تستخدم أنماط النعت التي سبق تعريفها كلما أمكن ذلك. ويمكن تعريف أنماط نعت إضافية على النحو المطلوب بالمواصفات الفردية.

ويجب أن توفر الاستخدامات الفردية لهذه التوصية خارطة تقابُل بين نموذج المعلومات هذا والهيكل الفعلي لقاعدة البيانات. ويسهل رسم خارطة التقابُل هذه إذا حُفظت المعلومات في دليل يعمل بالبروتوكول الخفيف للنفاذ إلى الدليل (LDAP) أو في دليل آمن قائم على المواصفات الواردة في [سلسلة التوصيات ITU-T X.500].

5.6 مبادئ التحكم الأساسي بالنفاذ

يتمثل الغرض من هذه الفقرة في إلقاء نظرة شاملة على المبادئ العامة لإقامة التحكم في النفاذ إلى بيانات القياس الحيوي عن بُعد. إذ يعنى التحكم في النفاذ التقليدي في الغالب بحق النفاذ أو برفض النفاذ إلى البيانات على أساس بعض المعلمات الدائمة. وتنتهج هذه التوصية نهجاً موسعاً من خلال النظر أيضاً في جوانب الحاجة إلى المعرفة. ويتم ذلك بانتهاج نهج الخدمة على النحو المبين في الفقرة 3.6. ولتنفيذ عملية معينة، يحتاج النافذ إلى امتياز يسمح بالنفاذ إلى خدمة النفاذ ذات الصلة وامتياز يسمح بالنفاذ إلى المعلومات اللازمة.

ويشمل الامتياز النفاذ إلى كائنات واحد أو أكثر من أصناف الكائن التي يمكن أن تقتصر على بعض الكائنات المسماة. وقد يختلف نمط العملية التي يمكن القيام بها على اختلاف أصناف الكائن والكائنات المسماة.

ويمكن أن تشمل العمليات على الكائنات عملية على فرادى النعوت وقيم النعوت. وقد تختلف العمليات المسموح بها على اختلاف أنماط النعوت.

6.6 العلاقة مع مخططات أخرى للتحكم في النفاذ

تختلف أنماط التحكم في النفاذ على اختلاف متطلبات التحكم في النفاذ. والقصد هنا هو إيراد وصف وجيز لمخططات مختلفة للتحكم في النفاذ وربطها بقائمة التحكم في النفاذ إلى بيانات القياس الحيوي عن بُعد (ACT).

1.6.6 التحكم الأساسي في النفاذ على النحو المعرَّف في التوصية [ITU-T X.501]

يُستخدم التحكم الأساسي في النفاذ على النحو المعرَّف في التوصية [ITU-T X.501] لحماية المعلومات في دليل على النحو المعرَّف في [سلسلة التوصيات ITU-T X.500]. وهو يوفر قوائم تحكم بالنفاذ مطولة تحدد تفاصيل كيف وإلى ماذا يُسمح لمختلف المستخدمين بالنفاذ. وهو لا يختلف عن هذه التوصية إلا بتلبية الحق بالمعرفة، ولكن ليس الحاجة للمعرفة.

2.6.6 التحكم في النفاذ القائم على القاعدة

تعرف كلتا التوصيتين [ITU-T X.501] و[b-ITU-T X.841] نمط التحكم في النفاذ القائم على القاعدة. وفي هذا النمط من التحكم في النفاذ، توسم البيانات المزمع حمايتها بمعلومات عن ماهية الحماية المطلوبة، في حين يمتلك المستخدمون النافذون المعلومات المعتمدة المرتبطة في طلب نفاذ يحدد مستوى التصريح لديهم فيما يتعلق بالنفاذ إلى معلومات معينة. ويختلف هذا المفهوم عن هذه التوصية بتطلُّب وسم بنود البيانات المخزنة، وبالاقتصار على تلبية حق المعرفة، دون تلبية الحاجة إلى المعرفة.

3.6.6 التحكم في النفاذ القائم على الدور على النحو المعرَّف للشبكات الذكية

إن التحكم في النفاذ القائم على الدور على النحو المعرَّف في المرجع [b-IEC 62351-8] يركز على المستخدمين وعلى الوظائف التي يؤديها المستخدمون. والدور هو مجموعة من حقوق الكائنات (الإجراءات التي يمكن القيام بها على أهداف معينة). ويمكن لمستخدم أن يتولى دوراً أو عدة أدوار. وفي التحكم في النفاذ، الدور هو نوع من الوسيط الذي يقلل من كمية المعلومات المطلوبة في قائمة التحكم بالنفاذ (ACL) من خلال تقليل تشعبات معلومات التحكم في النفاذ. ولا تُسرد الأذونات لكائنات نظام لكل مستخدم على حدة، بل تُسنَد أدوار إلى المستخدمين وتوضح حقوق كل دور مرة واحدة فقط.

4.6.6 التحكم في النفاذ القائم على النعت

التحكم في النفاذ القائم على النعت (ABAC) هو نموذج منطقي للتحكم في النفاذ يمكن تمييزه لأنه يتحكم في النفاذ إلى الكائنات بتقييم القواعد قياساً بنعوت الكيانات (الفاعل ومن يقع عليه الفعل منها)، والعمليات والبيئة ذات الصلة بطلب. وتستطيع أنظمة التحكم في النفاذ القائم على النعت إنفاذ مفاهيم التحكم التقديري في النفاذ والتحكم الإلزامي في النفاذ على حد سواء. والتحكم في النفاذ القائم على النعت يمكِّن التحكم الدقيق في النفاذ الذي يسمح بعدد أكبر من المدخلات المنفصلة في قرار التحكم في النفاذ، موفراً مجموعة أكبر من التوليفات الممكنة من هذه المتغيرات لتعبر عن مجموعة أكبر وأدق من القواعد الممكنة لصياغة السياسات.

وللاطلاع على مقدمة جيدة للتحكم في النفاذ القائم على النعت، انظر المرجع [b-NIST 800-162].

7.6 نظرة عامة على البروتوكولات

1.7.6 بروتوكول تقييم الامتياز

يتألف بروتوكول تقييم الامتياز من مجموعة من أنماط محتوى قواعد تركيب الرسالة التجفيرية (CMS). ويتطلب كل نمط من النفاذ نمط محتوى الطلب ونمط محتوى النتيجة.

وترسَل حالة نمط المحتوى باستخدام قواعد تركيب الرسالة التجفيرية (CMS) كما ترد ملفات تعريفها في الملحق B. وترد في الملحق C وحدة ASN.1 الرسمية.

2.7.6 بروتوكول تخصيص الامتياز

يُستخدم بروتوكول تخصيص الامتياز لإرسال شهادات النعت سواء بين مصدري سلطة أو من مصدر سلطة إلى حامل الامتياز.

ويعرَّف زوج واحد من أنماط المحتوى لهذا البروتوكول: نمط محتوى لتسيير الامتياز في شكل شهادات نعت ونمط محتوى للإبلاغ عن النتيجة.

وترسَل حالة نمط المحتوى باستخدام قواعد تركيب الرسالة التجفيرية (CMS) كما ترد ملفات تعريفها في الملحق B. وترد في الملحق C وحدة ASN.1 الرسمية.

8.6 استخدام قواعد تركيب الرسالة التجفيرية (CMS)

يرد في الملحق B ملف تعريف استخدام بيانات القياس الحيوي عن بُعد.

ولضمان الاستيقان من مصدر المعلومات بشكل صحيح، يجب أن تكون محتويات الأنماط المحددة بهذه التوصية مغلفة في حالة من نمط المحتوى **signedData**. ونظراً لإمكانية إرسال معلومات حساسة، يوصى أيضاً بالتغليف في حالة من نمط المحتوى **envelopedData**. ولم يُستخدم نمط المحتوى **ct-authEnvelopedData** في هذه التوصية.

9.6 اعتبارات شهادة المفتاح العمومي

يتعين أن توقَّع شهادة نعت من المصدِر باستخدام مفتاحه الخاص على أن يُتحقق منها باستخدام شهادة المفتاح العمومي المقابلة الصادرة لمصدِر شهادة النعت.

ومن حيث المبدأ، يمكن استخدام المفتاح الخاص نفسه للتوقيع على رسالة قواعد تركيب الرسالة التجفيرية (CMS) باستخدام نمط المحتوى **signedData**. ومن شأن ذلك أن يسهل عملية التحقق. بيد أن هناك مخاوف أمنية من استخدام المفتاح الخاص نفسه لأغراض مختلفة. وينبغي النظر في استخدام مفاتيح خاصة مختلفة للغرضين، ولكن هذه التوصية لا تتطلب ذلك بصفة إلزامية.

7 تقديم معلومات الامتياز

1.7 استخدام شهادات النعت

تتيح التوصية [ITU-T X.509] لشهادات المفتاح العمومي وشهادات النعت على السواء حمل معلومات امتياز. وفي كلتا الحالتين، تُحمل مواصفات الامتياز في نعوت على النحو المحدد في التوصية [ITU-T X.501]. وتتمايز أنماط النعت لهذا الغرض عن أنماط النعوت المستخدمة لنمذجة معلومات عن الكائنات. وبينما ينبغي أن تبقى أنماط النعوت المحددة لحمل المعلومات عن الكائنات بسيطة بقدر الإمكان، ستكون أنماط النعوت اللازمة لحمل معلومات امتياز معقدة بطبيعتها نوعاً ما.

وعند استخدام شهادة نعت لحمل معلومات امتياز:

أ) يحدد مكون **الحامل** (**holder**) الكيان الذي تخصَص له الامتيازات. وعندما يقدم حامل امتياز شهادة النعت هذه الحاملة للامتيازات إلى مدقق الامتيازات، يتعين أن يقوم مدقق الامتيازات بالاستيقان من النافذ للتحقق من أن النافذ هو في الواقع حامل امتياز شهادة النعت؛

ب) ويتعين على مكون **المصدِر** (**issuer**) أن يحمل اسم مصدر السلطة أو اسم سلطة النعت (AA) التي انتُدبت لتخصيص امتيازات. ويتعين على مدقق الامتيازات أيضاً الحصول على صحة شهادة المفتاح العمومي للمصدِر والتحقق منها للتحقق من صحة التوقيع على شهادة النعت؛

ج) ويتعين على مكون **النعوت** (**attributes**) أن يحمل نعتاً من نمط **accessService**، على النحو المحدد في الفقرة 3.7.

ويتعين أن تُمهر شهادة النعت بتوقيع مصدر السلطة التي وافقت على الامتياز أو توقيع اسم سلطة النعت التي فُوضت بإصدار الشهادة.

وسيتبسط التحقق إذا كان لدى الحامل والمصدِر شهادات مفتاح عمومي صادرة عن نفس سلطة إصدار الشهادات (CA).

2.7 استخدام شهادات المفتاح العمومي

عند استخدام شهادة المفتاح العمومي لحمل معلومات امتياز:

أ) يحدد مكون **الفاعل** (**subject**) النافذَ الذي خُصصت له الامتيازات. وعندما يقدم نافذٌ شهادة المفتاح العمومي هذه الحاملة لامتيازات إلى مدقق الامتيازات، يقوم مدقق الامتيازات بالاستيقان من النافذ؛

ب) ويتعين على مكون **المصدِر** (**issuer**) أن يحمل اسم سلطة إصدار الشهادات (CA) المسؤولة عن إصدار شهادة المفتاح العمومي؛

ج) ويتعين أن تحمل التوسعة **subjectDirectoryAttributes** حالة نمط النعت **accessService** (انظر الفقرة 3.7).

3.7 نمط نعت خدمة النفاذ

1.3.7 قواعد تركيب نعت خدمة النفاذ

القصد من نمط نعت **خدمة النفاذ** (**accessService**) هو أن يُدرج في مكون **نعوت** (**attributes**) لشهادة نعت أو في التوسعة **subjectDirectoryAttributes** لشهادة مفتاح عمومي. وله قواعد التركيب التالية:

```
AccessService ATTRIBUTE ::= {
  WITH SYNTAX  AccessService
  ID           id-at-accessService }
```

يوفر نعت من نمط **AccessService** معلومات خاصة بالامتياز للسماح للمتحقق من الامتياز بالتحقق مما إذا كان ينبغي الوفاء بطلب النفاذ أم لا. وهو نمط نعت متعدد القيم يسمح بأنماط خدمة النفاذ المتعدد وبإدراج الأذون المرتبطة به في نفس النعت.

ولا يمكن لكيان ما استعمال خدمة غير مدرجة في هذا النعت.

ولنمط النعت **AccessService** قواعد التركيب التالية:

```
AccessService ::= SEQUENCE {
  serviceId          OBJECT IDENTIFIER,
  objectDef          SEQUENCE SIZE (1..MAX) OF ObjectSel,
  ... }
```

أما مكونات قيمة نمط بيانات **AccessService** فهي:

أ) المكون **serviceId** الذي يتعين أن يحدد نمط خدمة النفاذ التي يحظى فيها النافذ بامتياز ما؛

ب) المكون **objectDef** الذي يتعين أن يحدد أصناف الكائن الذي يخصَص من أجله امتياز لحامل شهادة النعت في نمط خدمة معين. فيجب أن يمتلك عنصراً واحداً لكل صنف كائن يخصَص من أجله امتياز. ولا يملك حامل امتياز في خدمة النفاذ هذه أي نفاذ إلى أصناف الكائن التي لم يدرجها المكون **objectDef**.

2.3.7 اختيار الكائنات

يوصَّف اختيار الكائنات التي يحظى فيها النافذ بامتيازات بحالة نمط البيانات **ObjectSel**.

```
ObjectSel ::= SEQUENCE {
  objecClass         OBJECT-CLASS.&id,
  objSelect          CHOICE {
    allObj       [0]  TargetSelect,
    objectNames  [1]  SEQUENCE SIZE (1..MAX) OF SEQUENCE {
      object            CHOICE {
        names        [1]  SEQUENCE SIZE (1..MAX) OF DistinguishedName,
        subtree      [2]  DistinguishedName,
        ... },
      select          TargetSelect,
      ... },
    ... },
  ... }
```

تحدد قيمة نمط البيانات **ObjectSel** الامتياز لكل صنف كائن خُصص بامتياز في نمط الخدمة التي يجري النفاذ إليها. ولها المكونات التالية:

- المكون **objectClass** سيوصِّف صنف الكائن الذي تُخصص له امتيازات؛
- المكون **objSelect** سيوصِّف ماهية الكائنات في صنف الكائن الذي يُخصص له امتياز. وله خياران:

أ) يتعين أن يؤخذ بالخيار **allObj** إذا سرت الامتيازات المخصصة بالتساوي على جميع كائنات الصنف. ويتعين أن يختزن حالة نمط البيانات **TargetSelect**؛

ب) يتعين أن يؤخذ بالخيار **objectNames** إذا سرى امتياز حصراً على كائنات مختارة من صنف الكائن المحدد. ويمكن أن يتألف من عناصر متعددة، حيث يتضمن كل عنصر المكونات التالية:

'1' مكون **الكائن** (**object**) سيوصِّف واحد أو أكثر من الكائنات التي تُخصص لها امتيازات. وله الخياران التاليان:

- خيار **الأسماء** (**names**) سيوصِّف اسم واحد أو أكثر من الكائنات التي تسري عليها الامتيازات؛
- خيار **الشجرة الفرعية** (**subtree**) يتعين أن يؤخذ به لمجموعة من الكائنات حيث لكل كائن اسم مميز يساوي الاسم المميز المختزَن في هذا الخيار أو له مكونات اسم أولية مساوية لتلك الموجودة في الاسم المميز؛

'2' ويتعين أن يختزن المكون المختار حالة نمط البيانات **TargetSelect**.

ولنمط البيانات **TargetSelect** قواعد التركيب التالية:

```
TargetSelect ::= SEQUENCE {
  objOper   ObjectOperations OPTIONAL,
  attrSel   AttributeSel     OPTIONAL,
  ... }
  (WITH COMPONENTS {..., objOper  PRESENT } |
   WITH COMPONENTS {..., attrSel  PRESENT } )
```

ولنمط البيانات **TargetSelect** المكونان الاختياريان التاليان، حيث يجب أن يكون أحدهما على الأقل موجوداً:

أ) المكون **objOper**، عندما يكون موجوداً، سيحدد العمليات المسموح بها على كائنات صنف الكائن أو الكائنات المختارة. وإن لم يكن موجوداً، لا يسمح بأي عمليات على الكائنات ككل؛

ب) المكون **attrSel**، عندما يكون موجوداً، يتعين أن يختزن قيمة نمط البيانات **AttributeSel** (انظر الفقرة 3.3.7). وإن لم يكن هذا المكون موجوداً، لا يسمح بأي عمليات على نعوت الكائنات.

3.3.7 اختيار أنماط النعت

```
AttributeSel ::= SEQUENCE {
  attSelect           CHOICE {
    allAttr        [0]  SEQUENCE {
      attrOper1      [0]  AttributeOperations OPTIONAL,
      ... },
    attributes     [1]  SEQUENCE SIZE (1..MAX) OF SEQUENCE {
      select              SEQUENCE SIZE (1..MAX) OF ATTRIBUTE.&id,
      attrOper2      [0]  AttributeOperations OPTIONAL,
      ... },
    ... },
  ... }
```

يوصِّف نمط البيانات **AttributeSel** أنماط النعت التي يسري عليها الامتياز. وهو يحتوي على المكون التالي:

– للمكون **attSelect** خياران:

أ) يتعين اختيار الخيار **allAttr** إذا سرى الامتياز على جميع نعوت الكائن (الكائنات). وهو يحتوي على المكون التالي:

'1' المكون **attrOper1** سيحدد العمليات التي يمكن إجراؤها على النعوت؛

ب) يتعين أن يؤخذ بخيار **النعوت** (**attributes**) إذا اقتصر سريان امتياز على بعض نعوت الكائن (الكائنات). وليس للنافذ امتياز في أنماط النعت غير المدرجة، ويتعين ألا يُعلم بأنماط النعت غير المدرجة هذه. ولهذا الخيار المكونان التاليان:

'1' مكون **الاختيار** (**select**) سيوصِّف واحداً أو أكثر من أنماط النعت التي تسري عليها الامتيازات؛

'2' المكون **attrOper2** سيوصِّف العمليات التي يمكن إجراؤها على النعوت.

4.7 العمليات على الكائنات ككل

يُستخدم نمط البيانات التالي لتحديد العمليات المسموح بها ضد كائن:

```
ObjectOperations ::= BIT STRING {
  read             (0),
  add              (1),
  modify           (2),
  delete           (3),
  rename           (4),
  discloseOnError (5) }
```

يتعين ضبط إذن **القراءة** (**read**) على النافذ المسموح له قراءة المعلومات من كائن.

ويتعين ضبط إذن **الإضافة** (**add**) على النافذ المسموح له إضافة كائنات جديدة من صنف كائن محدد. وهو يتطلب إذن **الإضافة** (**add**) لجميع الكائنات من صنف محدد. ولكل نعت يضاف إلى الكائن، يتعين منح إذن **الإضافة** (**add**) لنمط النعت (انظر الفقرة 5.7).

ويتعين ضبط إذن **التعديل** (**modify**) على النافذ المسموح له تعديل كائن قائم. ويتعين على النافذ أن يملك إذن **التعديل** (**modify**) كي يتاح تعديل صنف الكائن ككل أو تعديل كائن مسمى (كائنات مسماة). وإذا أضاف نافذ نعوتاً، يتعين عليه أن يملك إذن إضافة (**add**) لأنماط النعت المعنية وإذا حذف نافذ نعوتاً، يتعين عليه أن يملك إذن **حذف** (**delete**) لأنماط النعت المعنية. وإذا عدَّل نافذ نعوتاً، يتعين عليه أن يملك إذن **تعديل** (**modify**) لأنماط النعت المعنية. وإذا حذف نافذ نعوتاً، يتعين عليه أن يملك إذن **deleteValue** لأنماط النعت المعنية. وإذا استعاض نافذ عن نعوت، يتعين عليه أن يملك إذن **replaceAttribute** لأنماط النعت المعنية.

ويتعين ضبط إذن **الحذف** (**delete**) على النافذ المسموح له حذف كائن قائم. ويتعين على النافذ أن يملك إذن **الحذف** (**delete**) كي يتاح حذف صنف الكائن ككل أو حذف كائن مسمى (كائنات مسماة).

ويتعين ضبط إذن **إعادة تسمية** (**rename**) على النافذ المسموح له إعادة تسمية كائن قائم (كائنات قائمة). ويتعين على النافذ أن يملك إذن **إعادة تسمية** (**rename**) كي تتاح إعادة تسمية صنف الكائن ككل أو إعادة تسمية كائن مسمى (كائنات مسماة).

ويتعين ضبط **discloseOnError** على النافذ المسموح له معرفة وجود الكائن عندما تتعطل العملية.

5.7 العمليات على النعوت

```
AttributeOperations ::= BIT STRING {
  read              (0),
  compare           (1),
  add               (2),
  modify            (3),
  delete            (4),
```

```
  deleteValue      (5),
  replaceAttribute (6),
  discloseOnError  (7) }
```

يتعين ضبط إذن **القراءة** (**read**) لكل من أنماط النعت المطلوبة على النافذ المسموح له قراءة مثل هذه النعوت. ويتعين على النافذ أن يملك إذن **قراءة** (**read**) صنف الكائن ذي الصلة ككل أو الكائنات المسماة ذات الصلة. وبالإضافة إلى ذلك، يتعين عليه أن يملك إذن **قراءة** (**read**) جميع نعوت أصناف الكائن هذه، أو يتعين عليه أن يملك النفاذ إلى نمط النعت ذي الصلة (أنماط النعت ذات الصلة).

ويتعين ضبط إذن **المقارنة** (**compare**) على النافذ المسموح له مقارنة واحد أو أكثر من النعوت. ويتعين على الامتياز أن يملك إذن **قراءة** (**read**) صنف الكائن ككل أو الكائن المسمى (الكائنات المسماة). وبالإضافة إلى ذلك، يجب أن يمكنه النفاذ بإذن المقارنة إلى جميع نعوت أصناف الكائن المتاحة أو أن يقارن الإذن بنمط النعت ذي الصلة (أنماط النعت ذات الصلة).

ويتعين ضبط إذن **الإضافة** (**add**) على النافذ المسموح له إضافة نعت واحد أو أكثر. ويتعين على النافذ أن يملك إذن **التعديل** (**modify**) لصنف الكائن ككل أو لكائن مسمى (كائنات مسماة). وبالإضافة إلى ذلك، يتعين عليه أن يملك إذن **الإضافة** (**add**) إلى أنماط نعوت ذات الصلة.

ويتعين ضبط إذن **التعديل** (**modify**) على النافذ المسموح له تعديل نعت من نمط معين. وبالإضافة إلى ذلك، يتعين على النافذ أن يملك إذن **تعديل** (**modify**) صنف الكائن ككل أو تعديل كائن مسمى (كائنات مسماة).

ويتعين ضبط إذن **الحذف** (**delete**) على النافذ المسموح له حذف نعت واحد أو أكثر. وبالإضافة إلى ذلك، يتعين على النافذ أن يملك إذن **تعديل** (**modify**) صنف الكائن ككل أو تعديل كائن مسمى (كائنات مسماة).

ويتعين ضبط الإذن **deleteValue**، على النافذ المسموح له حذف قيمة نعت واحدة أو أكثر من نمط (أنماط) النعت. وبالإضافة إلى ذلك، يتعين على النافذ أن يملك إذن **تعديل** (**modify**) صنف الكائن ككل أو تعديل كائن مسمى (كائنات مسماة).

ويتعين ضبط الإذن **replaceAttribute** على النافذ المسموح له أن يستعيض عن نعت من نمط معين بنعت من نفس النمط. وبالإضافة إلى ذلك، يتعين على النافذ أن يملك إذن **تعديل** (**modify**) صنف الكائن ككل أو تعديل كائن مسمى (كائنات مسماة).

ويتعين ضبط الإذن **discloseOnError** على النافذ المسموح له معرفة وجود نعت عندما تتعطل العملية. وبالإضافة إلى ذلك، يسري الإذن **discloseOnError** على الكائن ككل.

6.7 معالجة الخطأ

يمكن أن تظهر أخطاء نتيجة لاستخدام قواعد تركيب الرسالة التجفيرية (CMS) كما يرد بحثه في الملحق A.5. وعندما يُكشف الخطأ، لا ضرورة لمزيد من التدقيق. ويجاب طي نتيجة طلب النفاذ بوجود خطأ في قواعد تركيب الرسالة التجفيرية (CMS).

ويمكن أن تظهر أخطاء أيضاً نتيجة طلب النفاذ الفعلي.

ويبلَّغ عن الخطأ بواسطة حالة نمط البيانات **AccessdErr** المعرَّفة في الفقرة 10.8.

8 بروتوكول تأكيد الامتياز

1.8 نظرة عامة

تتضمن مجموعة كائنات المعلومات التالية جميع أنماط المحتوى المعرَّفة ممثلةً بكائنات المعلومات التي تعرفها هذه التوصية.

```
ActContentTypes CONTENT-TYPE ::= {
    privAssignRequest |
    privAssignResult |
    readRequest |
    readResult |
```

```
    compareRequest |
    compareResult |
    addRequest |
    addResult |
    deleteRequest |
    deleteResult |
    modifyRequest |
    modifyResult |
    renameRequest |
    renameResult,
    ... }
```

وتشكل أنماط المحتوى المعرَّفة بمجموعة **ActContentTypes** بروتوكول تأكيد الامتياز الذي يضم عدداً من عمليات النفاذ المختلفة على النحو المحدد في الفقرات من 4.8 إلى 9.8.

2.8 مكونات الطلب المشتركة

ترد المكونات التالية في جميع الطلبات:

```
CommonReqComp ::= SEQUENCE {
  attrCerts [31]  AttributeCertificates OPTIONAL,
  serviceId [30]  OBJECT IDENTIFIER,
  invokId   [29]  INTEGER,
  ... }

AtributeCertificates ::= SEQUENCE SIZE (1..MAX) OF AttributeCertificate
```

وفيما يلي معلمات الطلب المشتركة:

أ) المكون **attrCert** ، عندما يكون موجوداً، سيحدد شهادة النعت أو مسير شهادة النعت التي تحمل الامتياز للنافذ. وإذا كان هذا المكون غائباً، يتعين أن يورَّد الامتياز للنافذ في شهادة المفتاح العمومي للكيان النهائي؛

ب) المكون **serviceId** سيحدد نمط الخدمة المزمع استحضاره؛

ج) المكون **invokId** يتعين أن يتخذ قيمة الصفر لأول عملية تُستحضر وبعد ذلك يزاد بواحد لكل عملية تُستحضر لاحقاً. وينبغي أن يملك مدى يضمن عدم معاودة استخدام نفس القيمة لفترة زمنية طويلة من أجل الاتصال بين كيانين. ويتعين أن يُنص على كشف الطلبات الناقصة وكشف الهجمات الجوابية. ويتعين على متلقي طلب استخدام نفس القيمة في الرد للسماح للنافذ بإقران نتيجة بالطلب المقابل.

3.8 النفاذ إلى خدمة

جميع أنماط العمليات على النحو الذي توصَّف به في الفقرات من 4.8 إلى 9.8 تتطلب النفاذ إلى خدمة معينة. ويتعين أن يقوم مدقق الامتياز (المتلقي) بالتحقق من أن الامتياز المخصص لنافذ، ضمن ما يرتبط به من شهادة النعت أو شهادة المفتاح العمومي، يسمح بالنفاذ إلى الخدمة المطلوبة.

ويتعين الرد بشفرة الخطأ **noSuchService** إن لم يملك النافذ الإذن باستدعاء نمط الخدمة المطلوبة أو إن لم يدعم مقدم الخدمة نمط الخدمة المطلوبة.

وإذا امتلك النافذ إذناً باستدعاء الخدمة، يجب التحقق مما إذا كانت العملية المطلوبة تتسق مع نمط الخدمة وإذا تعذر ذلك، تعين الرد بشفرة الخطأ **invalidOperationForService**.

4.8 عملية القراءة

تشمل عملية القراءة طلب قراءة ونتيجة قراءة مقابلة.

يُحمل طلب قراءة كحالة نمط المحتوى **readRequest** وتُحمل نتيجة القراءة كحالة نمط المحتوى **readResult**.

```
readRequest CONTENT-TYPE ::= {
              ReadRequest
IDENTIFIED BY id-readRequest
```

ويستخدم النافذ نمط المحتوى **readRequest** لقراءة المعلومات عن كائن معين.

```
ReadRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  selection [2] InformationSelection,
  ... }
```

يحدد نمط البيانات **ReadRequest** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للكائن الذي تُطلب المعلومات عنه؛

ب) مكون **الاختيار** (**selection**) سيحدد نمط المعلومات التي يطلبها النافذ (انظر الفقرة 11.8).

ويجب أن يفشل طلب القراءة إذا حدد الطلب كائناً غير معروف، ويتعين الرد عليه بشفرة الخطأ **noSuchObject**.

ويجب أن يفشل طلب القراءة إن لم يملك النافذ إذن **قراءة** (**read**) الكائن وفقاً للامتياز المخصص للنافذ. وإذا لم يُمنح إذن **القراءة** (**read**) يتعين الرد بشفرة الخطأ **insufficientAccessRight** إن كان لدى النافذ إذن **discloseOnError** للكائن، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchObject**.

ويلزم إذن **القراءة** (**read**) لنمط نعت في كل نعت يرد في الجواب. وإذا لم يملك النافذ إذن **قراءة** (**read**) نمط نعت معين، لا يرد نعت من هذا النمط في النتيجة. وإذا لم ترد أي نعوت في النتيجة، يفشل الطلب. وإذا ملك النافذ الإذن **discloseOnError** لجميع طلبات النعت، عندئذ يتعين الرد بشفرة الخطأ **insufficientAccessRight**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noInformation**.

```
readResult CONTENT-TYPE ::= {
              ReadResult
IDENTIFIED BY id-readResult }
```

ويجب أن يقوم مدقق الامتياز باستخدام حالة نمط المحتوى **readResult** للرد إما بالمعلومات المطلوبة أو بالإبلاغ عن حال خطأ.

```
ReadResult ::= SEQUENCE {
  object    DistinguishedName,
  result    CHOICE {
    success   [0] ObjectInformation,
    failure   [1] AccessdErr,
    ... },
  ... }
```

ويحدد نمط البيانات **ReadResult** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيختزن اسم الكائن الذي طُلبت المعلومات عنه؛

ب) مكون النتيجة سيختزن نتيجة طلب القراءة. وله خياران:

- يتعين اختيار خيار **النجاح** (**success**) إذا كانت المعلومات سترد في الجواب وستختزن حالة نمط البيانات **ObjectInformation** (انظر الفقرة 12.8). وتشكل المعلومات الواردة في الرد تقاطعاً بين ما طلبه النافذ وماهية المعلومات التي يُسمح له باستخراجها؛
- يتعين اختيار خيار **الفشل** (**failure**) إذا كان يتعين الإبلاغ عن خطأ.

5.8 عملية المقارنة

تشمل عملية المقارنة طلب المقارنة ونتيجة مقارنة مقابلة.

يُحمل طلب المقارنة كحالة نمط المحتوى **compareRequest** وتُحمل نتيجة المقارنة كحالة نمط المحتوى **compareResult**.

```
compareRequest CONTENT-TYPE ::= {
            CompareRequest
IDENTIFIED BY id-compareRequest }
```

تُستخدم حالة نمط المحتوى **compareRequest** لمقارنة القيمة المزعومة المعروضة لنمط نعت معين مع قيمة نعت من نفس النمط تعود لكائن معين.

```
CompareRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  purported [2] AttributeValueAssertion,
  ... }
```

يحدد نمط البيانات **CompareRequest** المحتوى الفعلي، وله المكونان التاليان علاوة على المكونات المحددة في الفقرة 2.8:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للكائن الذي تقارَن معه قيمة النعت؛

ب) المكون **المزعوم** (**purported**) سيختزن توليفة نمط النعت وقيمة النعت التي يقارَن معها نعت من نفس النمط يختزنه الكائن المعني.

ويجب أن يفشل طلب المقارنة إذا حدد الطلب كائناً غير معروف، ويتعين الرد بشفرة الخطأ **noSuchObject**.

ويجب أن يفشل طلب المقارنة إن لم يملك النافذ إذن **قراءة** (**read**) الكائن وفقاً للامتياز المخصص للنافذ. وإذا لم يُمنح إذن **القراءة** (**read**) للكائن، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إن كان لدى النافذ إذن **discloseOnError** للكائن، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchObject**.

ويجب أن يفشل طلب المقارنة إن لم يملك النافذ إذن **المقارنة** (**compare**) لنمط النعت المعني. وإن كان لدى النافذ إذن **discloseOnError** لنمط النعت، عندئذ يتعين الرد بشفرة الخطأ **insufficientAccessRight**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noInformation**.

```
compareResult CONTENT-TYPE ::= {
            CompareResult
IDENTIFIED BY id-compareResult
```

ويجب أن يقوم مدقق الامتياز باستخدام حالة نمط المحتوى **compareResult** للرد إما بالمعلومات المطلوبة أو بالإبلاغ عن حال خطأ.

```
CompareResult ::= SEQUENCE {
  object    DistinguishedName,
  result    CHOICE {
    success   [0] CompareOK,
    failure   [1] AccessdErr,
    ... },
  ... }

CompareOK ::= SEQUENCE {
  matched        [0] BOOLEAN,
  matchedSubtype [1] BOOLEAN OPTIONAL,
  ... }
```

يحدد نمط البيانات **CompareResult** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للكائن الذي قُدم بشأنه طلب مقارنة؛

ب) مكون النتيجة سيختزن نتيجة طلب النفاذ. وله خياران:

- إذا اختير خيار **النجاح** (**success**) تعين الرد بحالة نمط البيانات **CompareOK** مع المكونات التالية:

'1' مكون **التطابق** (**matched**) سيتخذ قيمة **صح** (**TRUE**) إذا كان لنعت من نمط النعت أو أحد أنماطه الفرعية قيمة مساوية لتلك الواردة في الطلب. وبالإضافة إلى ذلك، يتعين أن يكون المكون **matchedSubtype** موجوداً وله قيمة **صح** (**TRUE**) ليتطابق نمط فرعي. وسيتخذ مكون **التطابق** (**matched**) قيمة **خطأ** (**FALSE**) في حال عدم التطابق لنمط النعت أو أحد أنماطه الفرعية؛

- ويتعين اختيار خيار **الفشل** (**failure**) إذا وجب الرد بالإبلاغ عن خطأ.

6.8 عملية الإضافة

تشمل عملية الإضافة طلب إضافة ونتيجة الإضافة المقابلة.

ويُحمل طلب الإضافة كحالة نمط المحتوى **addRequest** وتُحمل نتيجة الإضافة كحالة نمط المحتوى **addResult**.

```
addRequest CONTENT-TYPE ::= {
            AddRequest
IDENTIFIED BY id-addRequest }
```

وتُستخدم حالة نمط المحتوى **addRequest** لإضافة كائن جديد إلى نظام المعلومات.

```
AddRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  attr      [2] SEQUENCE SIZE (1..MAX) OF Attribute {{SupportedAttributes}}
                OPTIONAL,
  ... }
```

يحدد نمط البيانات **AddRequest** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للكائن الجديد الذي سيضاف؛

ب) المكون **attr**، عندما يكون موجوداً، يجب أن يختزن واحداً أو أكثر من النعوت التي سترتبط بالكائن الجديد.

وإن لم يملك النافذ إذن **الإضافة** (**add**) لصنف الكائن، يتعين الرد بشفرة الخطأ **insufficientAccessRight**.

وإذا كان كائن موجوداً مسبقاً بالاسم المميز المورَّد ولدى النافذ إذن **discloseOnError**، يتعين الرد بشفرة الخطأ **objectAlreadyExists**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**.

وإن لم يملك النافذ إذن **الإضافة** (**add**) كي تُدرج جميع النعوت مع الكائن، يفشل الطلب. وإذا كان لدى النافذ إذن **discloseOnError** لجميع أنماط النعت المدرجة، يتعين الرد بشفرة الخطأ **insufficientAccessRight**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noInformation**.

```
addResult CONTENT-TYPE ::= {
            AddResult
IDENTIFIED BY id-addResult }
```

ويجب أن يقوم مدقق الامتياز باستخدام حالة نمط المحتوى **addResult** للرد إما بالمعلومات المطلوبة أو بالإبلاغ عن حال خطأ.

```
AddResult ::= CHOICE {
  success   [0] NULL,
  failure   [1] AccessdErr,
  ... }
```

يحدد نمط البيانات **AddResult** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) يجب أن يؤخذ بخيار **النجاح** (**success**) إذا أضيف الكائن؛

ب) يجب أن يؤخذ بخيار **الفشل** (**failure**) إذا وجب الرد بالإبلاغ عن خطأ.

7.8 عملية الحذف

تشمل عملية الحذف طلب حذف ونتيجة الحذف المقابلة.

ويُحمل طلب الحذف كحالة نمط المحتوى **deleteRequest** وتُحمل نتيجة الحذف كحالة نمط المحتوى **deleteResult**.

```
deleteRequest CONTENT-TYPE ::= {
              DeleteRequest
IDENTIFIED BY id-deleteRequest }
```

وتُستخدم حالة نمط المحتوى **deleteRequest** لحذف كائن موجود من نظام المعلومات.

```
DeleteRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object        DistinguishedName,
  ... }
```

يحدد نمط البيانات **DeleteRequest** قواعد تركيب المحتوى الفعلي، وله المكون التالي:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للقيد الذي سيُحذف.

وفي حال عدم وجود الكائن المراد حذفه، عندئذ يتعين الرد بشفرة الخطأ **noSuchObject**.

وإن لم يكن لدى النافذ إذن **الحذف** (**delete**) لصنف الكائن، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إذا امتلك النافذ إذن **discloseOnError** للكائن، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchObject**.

```
deleteResult CONTENT-TYPE ::= {
              DeleteResult
IDENTIFIED BY id-deleteResult }
```

ويجب أن يقوم مدقق الامتياز باستخدام حالة نمط المحتوى **deleteResult** للرد إما بالمعلومات المطلوبة أو بالإبلاغ عن حال خطأ.

```
DeleteResult ::= CHOICE {
  success   [0] NULL,
  failure   [1] AccessdErr,
  ... }
```

ولحالة نمط البيانات **DeleteResult** خياران:

أ) يجب أن يؤخذ بخيار **النجاح** (**success**) إذا حُذف الكائن؛

ب) يجب أن يؤخذ بخيار **الفشل** (**failure**) إذا وجب الرد بالإبلاغ عن خطأ.

8.8 عملية التعديل

تشمل عملية التعديل طلب تعديل ونتيجة التعديل المقابلة.

ويُحمل طلب التعديل كحالة نمط المحتوى **modifyRequest** وتُحمل نتيجة التعديل كحالة نمط المحتوى **modifyResult**.

```
modifyRequest CONTENT-TYPE ::= {
              ModifyRequest
IDENTIFIED BY id-modifyRequest }
```

وتُستخدم حالة نمط المحتوى **modifyRequest** لتعديل كائن موجود.

```
ModifyRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
```

```
  object          DistinguishedName,
  changes         SEQUENCE SIZE (1..MAX) OF ObjectModification,
  select          InformationSelection,
  ... }

ObjectModification ::= CHOICE {
  addAttribute      [0]  Attribute{{SupportedAttributes}},
  deleteAttribute   [1]  AttributeType,
  addValues         [2]  Attribute{{SupportedAttributes}},
  deleteValues      [3]  Attribute{{SupportedAttributes}},
  replaceAttribute  [4]  Attribute{{SupportedAttributes}},
  ... }
```

يحدد نمط البيانات **ModifyRequest** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيختزن الاسم المميز للقيد الذي سيعدَّل؛

– في حال عدم وجود الكائن، يتعين الرد بشفرة الخطأ **noSuchObject**؛

– وإن لم يكن لدى النافذ إذن **تعديل** (**modify**) الكائن، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إذا امتلك النافذ إذن **discloseOnError** للكائن، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchObject**.

ب) مكون **التغييرات** (**changes**) سيختزن المعلومات اللازمة لتعديل نعت واحد أو أكثر:

– وسيختزن الخيار **addAttribute** نعتاً جديداً تُزمع إضافته:

'1' وإن لم يكن لدى النافذ إذن **الإضافة** (**add**) لنمط النعت، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'2' وإذا كان نعت من نمط موجود بالفعل، يتعين الرد بشفرة الخطأ **attributeAlreadyExists** إذا امتلك النافذ الإذن **discloseOnError** لنمط النعت، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

– وسيحدد الخيار **deleteAttribute** النعت الذي سيُحذف:

'1' لم يكن لدى النافذ إذن **الحذف** (**delete**) لنمط النعت، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'2' وفي حال عدم وجود نعت من ذلك النمط، يتعين الرد بشفرة الخطأ **noSuchAttribute**؛

– وسيحدد الخيار **addValues** هوية نعت قائم حسب نمط النعت. والقيم التي ستضاف إلى النعت هي القيم المدرجة في هذا الخيار.

'1' وإن لم يختزن الكائن أي نعت من النمط المعين، يتعين الرد بشفرة الخطأ **noSuchAttribute** إذا امتلك النافذ إذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'2' وإن لم يكن لدى النافذ الإذن **addValue**، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'3' وإذا جرت محاولة لإضافة قيمة موجودة بالفعل، يتعين الرد بشفرة الخطأ **attributeValueAlreadyExists** إذا امتلك النافذ الإذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

– وسيحدد الخيار **deleteValues** هوية نعت حسب نمط النعت. والقيم التي ستُحذف من النعت هي القيم المدرجة في هذا الخيار.

'1' وإن لم يختزن الكائن أي نعت من النمط المعين، يتعين الرد بشفرة الخطأ **noSuchAttribute** إذا امتلك النافذ إذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'2' وإن لم يكن لدى النافذ الإذن **deleteValue**، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إذا امتلك النافذ الإذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchAttributeValue**؛

'3' وإذا حاول النافذ حذف قيمة نعت لا وجود لها، يتعين الرد بشفرة الخطأ **noSuchAttributeValue**؛

– وسيستعيض الخيار **replaceAttribute** عن نعت موجود بنعت جديد من نفس النمط.

'1' وإن لم يختزن الكائن أي نعت من النمط المعين، يتعين الرد بشفرة الخطأ **noSuchAttribute** إذا امتلك النافذ إذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **insufficientAccessRight**؛

'2' وإن لم يكن لدى النافذ الإذن **replaceAttribute**، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إذا امتلك النافذ الإذن **discloseOnError**، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchAttribute**.

```
modifyResult CONTENT-TYPE ::= {
             ModifyResult
IDENTIFIED BY id-modifyResult }
```

ويجب أن يقوم مدقق الامتياز باستخدام حالة نمط المحتوى **modifyResult** للرد إما بالمعلومات المطلوبة أو بالإبلاغ عن حال خطأ.

```
ModifyResult ::= SEQUENCE {
  result    CHOICE {
    success   [0] ObjectInformation,
    failure   [1] AccessdErr,
    ... },
  ... }
```

ولحالة نمط البيانات **ModifyResult** خياران:

أ) يجب أن يؤخذ بخيار **النجاح** (**success**) إذا عُدل الكائن؛

ب) يجب أن يؤخذ بخيار **الفشل** (**failure**) إذا وجب الرد بالإبلاغ عن خطأ.

9.8 عملية إعادة تسمية كائن

تشمل عملية إعادة تسمية كائن طلب إعادة تسمية ونتيجة إعادة التسمية المقابلة.

ويُحمل طلب إعادة التسمية كحالة نمط المحتوى **renameRequest** وتُحمل نتيجة إعادة التسمية كحالة نمط المحتوى **renameResult**.

```
renameRequest CONTENT-TYPE ::= {
              RenameRequest
IDENTIFIED BY id-renameRequest }
```

وتُستخدم حالة نمط المحتوى **renameRequest** لتغيير اسم كائن موجود.

```
RenameRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object        DistinguishedName,
  new           DistinguishedName,
  ... }
```

يحدد نمط البيانات **RenameRequest** قواعد تركيب المحتوى الفعلي، وله المكونان التاليان:

أ) مكون **الكائن** (**object**) سيحدد الاسم المميز الحالي للكائن المزمع إعادة تسميته؛

ب) مكون **الجديد** (**new**) سيوفر الاسم المميز الجديد للكائن.

وفي حال عدم وجود الكائن الذي تراد إعادة تسميته، يتعين الرد بشفرة الخطأ **noSuchObject**.

وإن لم يكن لدى النافذ إذن **إعادة تسمية** (**rename**) كائن مسمى، يتعين الرد بشفرة الخطأ **insufficientAccessRight** إذا امتلك النافذ إذن **discloseOnError** للكائن المسمى، وخلاف ذلك، يتعين الرد بشفرة الخطأ **noSuchObject**.

```
renameResult CONTENT-TYPE ::= {
              RenameResult
IDENTIFIED BY id-renameResult }
RenameResult ::= SEQUENCE {
  result    CHOICE {
    success   [0] NULL,
    failure   [1] AccessdErr,
    ... },
  ... }
```

ولحالة نمط البيانات **RenameResult** خياران:

أ) يجب أن يؤخذ بخيار **النجاح** (**success**) إذا عُدل الكائن؛

ب) يجب أن يؤخذ بخيار **الفشل** (**failure**) إذا وجب الرد بالإبلاغ عن خطأ.

10.8 التعامل مع خطأ

عند حدوث ظرف استثناء أثناء التعامل مع أي طلب، يتعين على المتلقي الرد بشفرة خطأ بإدراج حالة نمط البيانات **AccessErr** في النتيجة.

```
AccessdErr ::= CHOICE {
  cmsErr     [0] CmsErr,
  ActErr   [1] PbactErr,
  ... }
```

ويتعين الأخذ بخيار **cmsErr** في حال حدوث ظرف استثناء أثناء تقييم أنماط المحتوى المعرَّفة بقواعد تركيب الرسالة التجفيرية (CMS) (انظر الفقرة A.5).

ويتعين الأخذ بخيار **pbactErr** في حال عدم كشف أي خطأ عند تقييم حالات أنماط محتوى قواعد تركيب الرسالة التجفيرية (CMS)، وكشف خطأ في حالة مغلَّفة لنمط محتوى معين في التوصية PBACT.

11.8 اختيار المعلومات

يُستخدم نمط البيانات **InformationSelection** لتحديد ما طُلب من المعلومات بطلب قراءة أو تعديل.

```
InformationSelection ::= SEQUENCE {
  attributes      CHOICE {
    allAttributes   [0]  NULL,
    select          [1]  SEQUENCE SIZE (1..MAX) OF ATTRIBUTE.&id,
    ... },
  infoTypes        ENUMERATED {
    attributeTypesOnly     (0),
    attributeTypeAndValues (1),
    ... },
  ...}
```

ولنمط البيانات **InformationSelection** المكونان التاليان:

أ) مكون **النعوت** (**attributes**) سيحدد ماهية النعوت التي ينبغي الرد بها. وله خياران:

– يجب أن يؤخذ بالخيار **allAttributes** إذا أراد النافذ كل المعلومات عن الكائن؛ أو

– يجب أن يؤخذ بخيار **الاختيار** (**select**) عندما لا تُطلب سوى مجموعة مختارة من النعوت؛

ب) لمكون **infoTypes** بنود التعداد التالية:

– يجب أن يؤخذ ببند التعداد **attributeTypesOnly** إذا تعين الرد بأنماط النعت فقط. وفي هذه الحالة، يجب على النافذ أن يملك إذن **قراءة** (*read*) نمط النعت وفقاً للامتياز الراهن. وإن لم يكن الحال كذلك، يزال نمط النعت من النتيجة. وإذا أدى ذلك إلى عدم إيراد أي معلومات في الرد، يفشل الطلب؛

– يجب أن يؤخذ ببند التعداد **attributeTypesAndValues** إذا تعين الرد بالنمط والقيم على السواء للامتياز الساري. وفي هذه الحالة، يجب على النافذ أن يملك إذن **قراءة** (*read*) نمط النعت وفقاً للامتياز الراهن. وإن لم يكن الحال كذلك، يزال نمط وقيمة النعت من النتيجة. وإذا أدى ذلك إلى عدم إيراد أي معلومات في الرد، يفشل الطلب.

12.8 المعلومات عن الكائن

عندما يراد الرد بمعلومات عن كائن، يتعين الرد بها كحالة نمط البيانات التالية:

```
ObjectInformation ::= SEQUENCE {
  object  DistinguishedName,
  info    CHOICE {
    attr    SET SIZE (1..MAX) OF Attribute {{SupportedAttributes}},
    type    SET SIZE (1..MAX) OF AttributeType },
  ... }
```

يجب أن يختزن مكون **الكائن** (**object**) الاسم المميز للكائن الذي يُرد بالمعلومات من أجله.

ويجب أن يختزن مكون **المعلومات** (**info**) مجموعة النعوت التي تختزن المعلومات المطلوبة أو مجموعة أنماط النعت.

وفي حال عدم إيراد أي معلومات في الرد، سيفشل الطلب.

13.8 شفرات الخطأ المعرَّفة

يرد هنا تعريف شفرات الخطأ لأنماط محتوى التوصية PBACT.

```
PbactErr ::= ENUMERATED {
  noSuchService,
  invalidOperationForService,
  insufficientAccessRight,
  noSuchObject,
  noSuchAttribute,
  noSuchAttributeValue,
  objectAlreadyExists,
  attributeAlreadyExists,
  attributeValueAlreadyExists,
  noInformation,
  ... }
```

أ) يجب الرد بشفرة الخطأ **noSuchService** إذا حدد النافذ خدمة لا إذن له بها ولا يسمح له بالعلم بها أو خدمة غير مدعومة؛

ب) يجب الرد بشفرة الخطأ **invalidOperationForService** إذا كانت العملية المطلوبة غير ذات صلة بالخدمة المطلوبة؛

ج) يجب الرد بشفرة الخطأ **insufficientAccessRight** عندما يطلب النافذ خدمة لا إذن له بها أو عندما يريد تنفيذ عملية لا إذن له بها وفقاً للخدمة التي يراد النفاذ إليها؛

د) يجب الرد بشفرة الخطأ **noSuchObject** إذا حاول نافذ النفاذ إلى كائن غير موجود أو إلى كائن لا إذن له به ولا يسمح له بالعلم بوجوده؛

هـ) يجب الرد بشفرة الخطأ **noSuchAttribute** إذا حاول نافذ النفاذ إلى نعت غير موجود أو إلى نعت لا إذن له به ولا يسمح له بالعلم بوجوده؛

و) يجب الرد بشفرة الخطأ **noSuchAttributeValue** إذا حاول نافذ النفاذ إلى قيمة نعت غير موجودة أو إلى قيمة نعت لا إذن له بها ولا يسمح له بالعلم بوجودها؛

ز) يجب الرد بشفرة الخطأ **objectAlreadyExists** إذا جرت محاولة لإضافة كائن ذي اسم مميز يساوي الاسم المميز لكائن موجود بالفعل شريطة أن يكون لدى النافذ إذن بالعلم بوجود هذا الكائن؛

ح) يجب الرد بشفرة الخطأ **attributeAlreadyExists** إذا جرت محاولة لإضافة نعت من نمط موجود بالفعل داخل الكائن المعني شريطة أن يكون لدى النافذ إذن بالعلم بوجود نمط النعت هذا داخل الكائن؛

ط) يجب الرد بشفرة الخطأ **attributeValueAlreadyExists** إذا جرت محاولة لإضافة نعت لكائن يختزن بالفعل نعتاً من نفس النمط شريطة أن يكون لدى النافذ إذن بالعلم بوجود هذا النعت؛

ي) يجب الرد بشفرة الخطأ **noInformation** إذا طُلبت معلومات، ولكنها إما غير متوفرة أو أن النافذ لا إذن لديه بالعلم بوجود تلك البيانات.

9 بروتوكول تخصيص الامتياز

1.9 مجال تطبيق البروتوكول

يستخدم بروتوكول تخصيص الامتياز لتخصيص امتيازات:

أ) من مصدر السلطة (SOA) إلى سلطة نعت (AA) متوسطة كي تتواصل الإنابة؛

ب) من مصدر السلطة مباشرة إلى كيان يستخدم الامتيازات المخصصة لتأكيد تلك الامتيازات؛

ج) من سلطة النعت مباشرة إلى كيان يستخدم الامتيازات المخصصة لتأكيد تلك الامتيازات؛

د) من سلطة نعت إلى سلطة نعت أخرى، عندما تقع سلطات نعت متعددة في المسير بين مصدر السلطة وحامل الامتياز.

ملاحظة – يوصى بتقصير مسير الإنابة قدر الإمكان.

2.9 أنماط المحتوى

يستخدم بروتوكول تخصيص الامتياز نمطي محتوى: نمط محتوى لتخصيص امتيازات ونمط محتوى لتأكيد التخصيص.

1.2.9 نمط محتوى طلب تخصيص الامتياز

يُحمل طلب تخصيص الامتياز في حالة نمط المحتوى **privAssignRequest**.

```
privAssignRequest CONTENT-TYPE ::= {
            PrivAssignRequest
IDENTIFIED BY id-privAssignRequest }
```

وتوصَّف قواعد تركيب المحتوى الفعلي بنمط البيانات التالي:

```
PrivAssignRequest ::= SEQUENCE {
  attrCerts   AttributeCertificates OPTIONAL,
  ... }
```

وليس لهذا النمط من البيانات سوى مكون واحد يختزن تسلسلاً من شهادات النعت. فإذا أُرسل طلب تخصيص امتياز من مصدر سلطة، يتكون التسلسل من شهادة نعت واحدة فقط. وإذا وُجدت سلطة نعت واحدة ما بين مصدر السلطة وحامل الامتياز، يجب أن يتضمن الطلب من سلطة النعت إلى حامل الامتياز شهادة النعت الصادرة عن مصدر السلطة وكذلك شهادة النعت الصادرة عن سلطة النعت. وتلزم شهادة نعت أخرى لكل سلطة نعت إضافية تقع بين مصدر السلطة وحامل الامتياز.

2.2.9 نمط محتوى نتيجة تخصيص الامتياز

تُحمل نتيجة تخصيص الامتياز في حالة نمط المحتوى **privAssignResult**.

```
privAssignResult CONTENT-TYPE ::= {
              PrivAssignResult
IDENTIFIED BY id-privAssignResult }
```

وتوصَّف قواعد تركيب المحتوى الفعلي بنمط البيانات التالي:

```
PrivAssignResult ::= SEQUENCE {
  result  CHOICE {
    success  NULL,
    failure  PrivAssignErr },
  ... }

PrivAssignErr ::= CHOICE {
--cmsErr     [0] CmsErr,
  assignErr  [1] AssignErr,
  ... }
```

3.2.9 شفرات الخطأ المعرَّفة

```
AssignErr ::= ENUMERATED {
  invalidAttributeCertificate (0),
  invalidDelegationPath
  invalidPublicKeyCertificate
    ... }
```

الملحق A

توزيع معرف الكائن في سلسلة التوصيات ITU-T 1080

(يشكل هذا الملحق جزءاً أساسياً من هذه التوصية)

1.A المستوى الأعلى لشجرة معرف الكائن

يوزع الملحق A بالتوصية [ITU-T X.1081] أقواساً تحت القوس الموزع لبيانات القياس الحيوي عن بُعد وهو:

```
id-telebio OBJECT IDENFIER ::= { joint-iso-itu-t(2) telebiometrics(42) }
```

وتحت هذا القوس، توزع التوصية [ITU-T 1081] القوس التالي للرعاية الصحية عن بُعد:

```
id-th OBJECT IDENTIFIER ::= { id-telebio th(3) }
```

وقد وزعت التوصية [ITU-T X.1080.1] عدة أقواس تحت قوس **id-th**. والقوس '0' موزع لوحدات ASN.1 المعرَّفة ضمن التوصية [ITU-T X.1080.1]. وقد وُزعت أقواس أخرى لفئات الكيان. وتوصِّف هذه التوصية أن قوس '0' يجب أن يُستخدم أيضاً لتوزيع معرفات الكائن في سلسلة التوصيات ITU-T X.1080 بشكل عام. ولتجنب التضارب، تُستخدم القيمة 10 لتوزيع معرفات الكائن في سلسلة التوصيات ITU-T X.1080.

```
id-telehelth OBJECT IDENTIFIER ::= { id-th all(0) telehealth(10) }
```

وتوزَّع الأقواس التالية لأجزاء مختلفة من سلسلة التوصيات ITU-T X.1080:

```
id-x1080-0 OBJECT IDENTIFIER ::= { id-telehelth part0(0) }
id-x1080-1 OBJECT IDENTIFIER ::= { id-telehelth part1(1) }
id-x1080-2 OBJECT IDENTIFIER ::= { id-telehelth part2(2) }
----
```

وينقسم قوس لجزء معين من سلسلة التوصيات ITU-T X.1080 على النحو التالي:

- وحدات لها القوس '0'؛
- أنماط محتوى قواعد تركيب الرسالة التجفيرية (CMS) لها القوس '1'؛
- أنماط النعت لها القوس '2'.

ويمكن لجزء معين أن يوزع أقواساً إضافية وفقاً لاحتياجاته.

وفي هذه التوصية، يوزَّع القوس التالي للوحدات:

```
id-x1080-0-module OBJECT IDENTIFIER ::= { id-x1080-0 module(0) }
```

ويوزَّع القوس التالي لأنماط محتوى قواعد تركيب الرسالة التجفيرية (CMS):

```
id-x1080-0-Cont   OBJECT IDENTIFIER ::= { id-x1080-0 cmsCont(1) }
```

ويوزَّع القوس التالي لأنماط النعت المستخدمة لتخصيص الامتيازات:

```
id- x1080-0-attr   OBJECT IDENTIFIER ::= { id-x1080-0 prAttr(2) }
```

2.A معرفات الكائن في أنماط محتوى قواعد تركيب الرسالة التجفيرية (CMS)

توزَّع معرفات الكائن التالية لأنماط المحتوى المعرَّفة لبروتوكول تخصيص الامتياز وبروتوكول تقييم الامتياز:

```
id-privAssignReq  OBJECT IDENTIFIER ::= { id-x1080-0-Cont privAssignRequest(1) }
id-privAssignRes  OBJECT IDENTIFIER ::= { id-x1080-0-Cont privAssignResult(2) }
id-readRequest    OBJECT IDENTIFIER ::= { id-x1080-0-Cont readRequest(3) }
id-readResult     OBJECT IDENTIFIER ::= { id-x1080-0-Cont readResult(4) }
id-compareRequest OBJECT IDENTIFIER ::= { id-x1080-0-Cont compareRequest(5) }
id-compareResult  OBJECT IDENTIFIER ::= { id-x1080-0-Cont compareResult(6) }
id-addRequest     OBJECT IDENTIFIER ::= { id-x1080-0-Cont addRequest(7) }
id-addResult      OBJECT IDENTIFIER ::= { id-x1080-0-Cont addResult(8) }
id-deleteRequest  OBJECT IDENTIFIER ::= { id-x1080-0-Cont deleteRequest(9) }
id-deleteResult   OBJECT IDENTIFIER ::= { id-x1080-0-Cont deleteResult(10) }
id-modifyRequest  OBJECT IDENTIFIER ::= { id-x1080-0-Cont modifyRequest(11) }
id-modifyResult   OBJECT IDENTIFIER ::= { id-x1080-0-Cont modifyResult(12) }
id-renameRequest  OBJECT IDENTIFIER ::= { id-x1080-0-Cont renameRequest(13) }
id-renameResult   OBJECT IDENTIFIER ::= { id-x1080-0-Cont renameResult(14) }
```

3.A معرفات الكائن في أنماط نعت الامتياز

```
id-at-accessSer   OBJECT IDENTIFIER ::= { id-pbactPrivAttr 1 }
```

الملحق B

ملف تعريف قواعد تركيب الرسالة التجفيرية

(يشكل هذا الملحق جزءاً أساسياً من هذه التوصية)

1.B اعتبارات عامة

يرد تعريف قواعد تركيب الرسالة التجفيرية (CMS) في المرجع [IETF RFC 5652]. وهو يعرف قدرات الاتصالات التي تتيح سلامة البيانات والاستيقان منها وكتمانها. وترد في المرجع [IETF RFC 5083] مواصفات إضافية. وترد في المرجعين [IETF RFC 5911] و[IETF RFC 6268] وحدات ASN.1 جديدة لقواعد تركيب الرسالة التجفيرية (CMS). ويقدم هذا الملحق ملف تعريف قواعد تركيب الرسالة التجفيرية (CMS) كي تستخدمه مواصفات بيانات القياس الحيوي عن بُعد بالإحالة إلى هذه المواصفات.

وقواعد تركيب الرسالة التجفيرية (CMS) هي مواصفات متعددة الاستعمالات معدة للاستخدام في العديد من البيئات المختلفة. ولا يستخدم ملف التعريف هذا كل قدرات قواعد تركيب الرسالة التجفيرية. ويشمل ملف التعريف هذا أنماط بيانات قواعد تركيب الرسالة التجفيرية المستخدمة في هذه التوصية وغيرها من مواصفات بيانات القياس الحيوي عن بُعد. ويجوز لهذه المواصفات الأخرى لبيانات القياس الحيوي عن بُعد أن تستشهد بهذا الملحق في استخدامها لقواعد تركيب الرسالة التجفيرية (CMS).

وليس القصد أن توضع مواصفة لتنفيذ لا يلتزم بمعايير IETF RFC، بل أن تناقَش تلك الجوانب من قواعد تركيب الرسالة التجفيرية (CMS) ذات الصلة بمواصفات بيانات القياس الحيوي عن بُعد. ويقدم التذييل I وحدة ASN.1 غير رسمية تعبر عن استخدام بيانات القياس الحيوي عن بُعد لقواعد تركيب الرسالة التجفيرية.

وتحدد قواعد تركيب الرسالة التجفيرية (CMS) أنماطاً مختلفة من المحتوى تُستخدم لأغراض مختلفة. ومواصفات بيانات القياس الحيوي عن بُعد تستخدم نمط المحتوى **signedData** لتقديم الاستيقان والسلامة، وتستخدم نمط المحتوى **envelopedData** لتقديم التجفير وبالتالي الكتمان، وتستخدم نمط المحتوى **ct-authEnvelopedData**. ويُستخدم نمط المحتوى **signedData** عندما يُتطلب التوقيع الرقمي، في حين يُستخدم نمط المحتوى **envelopedData** عندما يُتطلب الكتمان. وعند استخدام حالة نمط المحتوى **envelopedData** فهي تغلَّف في حالة نمط المحتوى **signedData**. ويُستخدم نمط المحتوى **ct-authEnvelopedData** عندما تشكل رسائل متعددة مهمة معينة.

ولا تستخدم مواصفات بيانات القياس الحيوي عن بُعد جميع جوانب أنماط المحتويات المذكورة أعلاه. لذا يقدم هذا الملحق ملف تعريف لاستخدام قواعد تركيب الرسالة التجفيرية (CMS) في بيانات القياس الحيوي عن بُعد. وتسهيلاً للمرجعية، ترد هنا الجوانب ذات الصلة بقواعد تركيب الرسالة التجفيرية.

وتغلَّف حالة نمط المحتوى معرَّفة بمواصفات بيانات القياس الحيوي عن بُعد في حالة نمط المحتوى **envelopedData**، إذا كان الكتمان مطلوباً؛ وخلاف ذلك، فهي تغلَّف في حالة نمط المحتوى **signedData**. وبدلاً من ذلك، يمكن إدراج حالة نمط محتوى كهذه في حالة نمط المحتوى **ct-authEnvelopedData**.

ويعرَّف نمط المحتوى وفق المرجع [IETF RFC 6268] باستخدام صنف كائن المعلومات التالي:

```
CONTENT-TYPE ::= TYPE-IDENTIFIER
```

وصنف كائن المعلومات **CONTENT-TYPE** يكافئ صنف كائن المعلومات **TYPE-IDENTIFIER** المنشأ بقواعد ASN.1. ويُستخدم كائن معلومات نمط المحتوى (**CONTENT-TYPE**) لربط نمط المحتوى الذي يحدده معرف كائن بقواعد تركيب المحتوى المجردة.

ويوفر نمط بيانات **ContentInfo** التالي قواعد التركيب العامة لنمط المحتوى:

```
ContentInfo ::= SEQUENCE {
  contentType  CONTENT-TYPE.&id ({TelebSupportedcontentTypes}),
  content      CONTENT-TYPE.&Type
         ({TelebSupportedcontentTypes}{@contentType})OPTIONAL,
  ... }

TelebSupportedcontentTypes CONTENT-TYPE ::=
  { signedData | envelopedData | ct-authEnvelopedData, ...}
```

أما أنماط المحتوى المدعومة فهي نمط المحتوى **signedData** ونمط المحتوى **envelopedData** ونمط المحتوى **ct-authEnvelopedData** ومجموعة أنماط المحتوى التي تعرِّفها مواصفة معينة لبيانات القياس الحيوي عن بُعد.

وتتطلب قواعد تركيب الرسالة التجفيرية (CMS) أن توصَّف نسخة قواعد تركيب الرسالة التجفيرية لنمط البيانات بحيث تبين قواعد التركيب المحددة المستخدمة لهذا النمط من البيانات. وتعرَّف النسخ التالية:

```
CMSVersion ::= INTEGER{ v0(0), v1(1), v2(2), v3(3), v4(4), v5(5) }
```

ويعرِّف المرجع [IETF RFC 6268] نمط البيانات التالي ذا المعلمات المستخدم في جميع فقرات المواصفات:

```
Attributes { ATTRIBUTE:AttrList } ::=
            SET SIZE (1..MAX) OF Attribute {{ AttrList }}
```

2.B استخدام نمط المحتوى signedData

يوصَّف نمط المحتوى التالي في الفقرة 5 من المرجع [IETF RFC 5652]. وباستخدام ترميز معدل بشكل طفيف يعبر عن استخدامه في بيانات القياس الحيوي عن بُعد، فهو يوصَّف على النحو التالي:

```
SignedData ::= SEQUENCE  {
  version               CMSVersion (v3),
  digestAlgorithms      SET (SIZE (1)) OF AlgorithmIdentifier
                          {{Teleb-Hash-Algorithms}},
  encapContentInfo      EncapsulatedContentInfo,
  certificates     [0]  IMPLICIT SET (SIZE (1..MAX)) OF Certificate OPTIONAL,
  crls             [1]  IMPLICIT RevocationInfoChoices OPTIONAL,
  signerInfos           SignerInfos,
  ... }
```

الملاحظة 1 – يستخدم المرجع [IETF RFC 6268] نسخة معدلة بعض الشيء من نمط البيانات **AlgorithmIdentifier**؛ بيد أن ملف التعريف هذا يستخدم نمط البيانات **AlgorithmIdentifier** على النحو المحدد في التوصية [ITU-T X.509].

ويتعين أن يتخذ مكون **النسخة** (**version**) القيمة **v3** وفقاً للفقرة 1.5 من المرجع [IETF RFC 5652].

ويتعين أن يتكون المكون **digestAlgorithms** من عنصر واحد يوصِّف خوارزمية الاختزال من مجموعة من خوارزميات الاختزال المطبقة على النحو المحدد في مواصفة بيانات القياس الحيوي عن بُعد المعمول بها.

ويسمح التعريف التالي بإدراج أي خوارزمية اختزال.

```
Teleb-Hash-Algorithms ALGORITHM ::= {...}
```

الملاحظة 2 – لا يفرض ملف التعريف هذا خوارزمية اختزال محددة. وتمكن الاستعاضة عن المواصفات المرجعية أو اتفاقات المنفذ بنقاط مع مجموعة من خوارزميات اختزال تُدعم في بيئة معينة.

الملاحظة 3 – تسمح قواعد تركيب الرسالة التجفيرية (CMS) بتوقيعات رقمية متعددة، ومن ثم بخوارزميات اختزال متعددة. ولا صلة للتوقيعات الرقمية المتعددة بمواصفات بيانات القياس الحيوي عن بُعد.

ويتعين أن يختزن المكون **encapContentInfo** حالة نمط البيانات التالية:

```
EncapsulatedContentInfo ::= SEQUENCE {
  eContentType      CONTENT-TYPE.&id({envelopedData, ...}),
  eContent     [0] EXPLICIT OCTET STRING
```

```
    (CONTAINING CONTENT-TYPE.&Type({envelopedData, ...}
     {@eContentType})) OPTIONAL }
```

ويحتوي هذا النمط من البيانات على المكونات التالية:

أ) المكون **eContentType** سيختزن معرف الكائن الذي يحدد هوية نمط المحتوى المغلف. ويتعين أن يختزن هذا المكون هوية نمط المحتوى **envelopedData**، إذا كان التجفير مطلوباً. وإذا لم يكن التجفير مطلوباً، فعليه أن يختزن واحداً من أنماط المحتوى التي توصِّفها مواصفات بيانات القياس الحيوي عن بُعد ذات الصلة؛

ب) ويتعين أن يختزن مكون **المحتوى الإلكتروني** (**econtent**) المحتوى المغلف الفعلي في سلسلة أثمونات. ويجب أن يكون حاضراً دائماً.

الملاحظة 4 – يعرَّف هذا المكون كمكون اختياري. غير أن لجميع أنماط المحتوى ذات الصلة محتوى محدداً.

ويتعين أن يختزن مكون **الشهادات** (**certificates**) شهادات المفتاح العمومي بما يكفي لإنشاء مسير واحد لمنح الشهادات على النحو الذي يحدده نمط البيانات **PkiPath** المعرَّف في التوصية [ITU-T X.509].

وليس لمكون **crls** صلة بمواصفات بيانات القياس الحيوي عن بُعد ويجب أن يكون غائباً.

ويتعين أن يختزن المكون **signerInfos** حالة نمط البيانات **SignerInfos**:

```
SignerInfos ::= SET (SIZE (1)) OF SignerInfo

SignerInfo ::= SEQUENCE {
  version                 CMSVersion (v1),
  sid                     SignerIdentifier,
  digestAlgorithm         AlgorithmIdentifier {{Teleb-Hash-Algorithms}},
  signedAttrs        [0]  IMPLICIT Attributes{{SignedAttributes}} OPTIONAL,
  signatureAlgorithm      AlgorithmIdentifier {{Teleb-Signature-Algorithms}},
  signature               SignatureValue,
  unsignedAttrs      [1]  IMPLICIT Attributes {{UnsignedAttributes}} OPTIONAL,
  ... }

SignerIdentifier ::= CHOICE {
  issuerAndSerialNumber    IssuerAndSerialNumber,
  subjectKeyIdentifier [0] SubjectKeyIdentifier,
  ...}

IssuerAndSerialNumber ::= SEQUENCE {
  issuer       Name,
  serialNumber CertificateSerialNumber }

SignedAttributes ATTRIBUTE ::= { contentType | messageDigest, ... }

Teleb-Signature-Algorithms ALGORITHM ::= {...}

SignatureValue ::= OCTET STRING

UnsignedAttributes ATTRIBUTE ::= {...}
```

لا يدعم ملف التعريف هذا سوى موقِّع واحد، لذلك يجب أن تملك حالة نمط البيانات **SignerInfos** عنصراً واحداً فقط لا غير. ولنمط البيانات **SignerInfo** المكونات التالية:

أ) مكون **النسخة** (**version**) سيحدد **v1** وفقاً للمرجع [IETF RFC 5652]؛

ب) المكون **sid** سيحدد هوية شهادة المفتاح العمومي للكيان النهائي لدى الموقِّع وسيختزن حالة نمط البيانات **SignerIdentifier**. ويحدد هذا النمط من البيانات خيارين

– يحدد الخيار **issuerAndSerialNumber** هوية شهادة المفتاح العمومي للكيان النهائي بتحديد الاسم المميز لسلطة إصدار الشهادات التي أصدرت الشهادة والرقم التسلسلي لشهادة المفتاح العمومي. ويجب دائماً أن يؤخذ بهذا الخيار؛

– ولا يجوز أن يؤخذ بالخيار **subjectKeyIdentifier**؛

ج) المكون **digestAlgorithm** سيتخذ نفس القيمة المستخدمة في مكون **digestAlgorithms** من نمط البيانات **SignerInfo**؛

د) المكون **signedAttrs** سيختزن قائمة النعوت الموقعة. ويتطلب المرجع [IETF RFC 5652] بالحد الأدنى إدراج حالات نمطي النعت **contentType** و**messageDigest**. ولا يتطلب ملف التعريف هذا إدراج أي نعت إضافي، ولكن مواصفات الإحالة المرجعية يمكن أن تضيف إلى القائمة؛

هـ) المكون **signatureAlgorithm** سيختزن خوارزمية التوقيع المستخدمة لإنشاء التوقيع الرقمي الذي يختزنه مكون **التوقيع** (**signature**)؛

الملاحظة 5 – لا يفرض ملف التعريف هذا خوارزمية توقيع محددة. وتمكن الاستعاضة عن المواصفات المرجعية أو اتفاقات المنفذ بنقاط مع مجموعة من خوارزميات توقيع تُدعم في مواصفة معينة لبيانات القياس الحيوي عن بُعد.

و) ويرد المكونان **signature** و**unsignedAttrs** على النحو المحدد في المرجع [IETF RFC 5652].

3.B استخدام نمط المحتوى envelopedData

1.3.B اعتبارات عامة

يسمح نمط المحتوى **envelopedData** بتجفير البيانات. وهذا يتطلب إنشاء مفاتيح متناظرة مشتركة. ويقدم المرجع [IETF RFC 5652] تقنيات مختلفة لتوليد مثل هذه المفاتيح المتناظرة. ويتطلب ملف التعريف هذا تقنية اتفاق المفتاح المعروفة بأسلوب اتفاق مفتاح ديفي هيلمان (DH). ويرد في المرجع [IETF RFC 2631] توصيف أسلوب اتفاق مفتاح ديفي هيلمان لتقنية منحنى غير إهليلجي. ويقدم المرجع [IETF RFC 5753] مواصفات لاستخدام تقنيات منحنى إهليلجي.

وينتج عن أسلوب ديفي هيلمان (DH) سر مشترك يمكن استخدامه كمادة مفتاحية تسمح بتوليد مفاتيح متناظرة مشتركة. ويعترف ملف التعريف هذا بوضعين لتشغيل أسلوب ديفي هيلمان (DH): الوضع سريع الزوال – الساكن والوضع الساكن – الساكن.

ويتطلب الوضع سريع الزوال – الساكن أن يمتلك المتلقي شهادة المفتاح العمومي مع المفتاح العمومي DH على النحو الذي تصادق عليه سلطة إصدار الشهادات. ويجب أن تتاح شهادة المفتاح العمومي هذه للمرسل. وينشئ المرسل زوجاً جديداً من مفاتيح DH لكل رسالة يرسلها. وبهذه الطريقة، يصبح السر المشترك مختلفاً في كل رسالة.

ويتطلب الوضع الساكن – الساكن أن يمتلك كلا الكيانين المتواصلين شهادة المفتاح العمومي DH المصدقة. وبما أن هذا الوضع من شأنه أن ينتج السر المشترك نفسه لكل رسالة، لابد من أن يورِّد المرسل للمستخدم مادة مفتاحية عشوائية ما للحصول على مادة مفتاحية مختلفة لكل رسالة.

ويتطلب ملف التعريف هذا أن يُدعم الوضع سريع الزوال – الساكن.

ويتطلب كلا الأسلوبين حصول كلا الكيانين المتواصلين على شهادة المفتاح العمومي DH المصدقة للكيان النهائي لدى شريكه، فيما تمضي الاتصالات في كلا الاتجاهين.

ويرد في الفقرة 6 من المرجع [IETF RFC 5652] توصيف نمط المحتوى التالي ويرد تحديثه في المرجع [IETF RFC 6268]:

```
envelopedData CONTENT-TYPE ::= {
                EnvelopedData
  IDENTIFIED BY id-envelopedData }
EnvelopedData ::= SEQUENCE {
  version                    CMSVersion(v0 | v2),
```

```
  originatorInfo          [0] IMPLICIT OriginatorInfo OPTIONAL,
  recipientInfos              RecipientInfos,
  encryptedContentInfo        EncryptedContentInfo,
  ...,
  [[2: unprotectedAttrs   [1] IMPLICIT Attributes
       {{UnprotectedAttributes}} OPTIONAL ]] }
UnprotectedAttributes ATTRIBUTE ::=
  { aa-CEKReference | aa-CEKMaxDecrypts | aa-KEKDerivationAlg }
```

وللمكون **EnvelopedData** المكونات التالية:

أ) مكون **النسخة** (**version**) يتعين أن يتخذ القيمة **v2** وفقاً للمرجع [IETF RFC 5652]. إذا كان المكون **unprotectedAttrs** موجوداً، وخلاف ذلك، يجب أن يتخذ القيمة **v0**؛

ب) المكون **originatorInfo** يجب أن يكون غائباً؛

ج) المكون **recipientInfos** سيختزن حالة نمط البيانات **RecipientInfos** على النحو المحدد في الفقرة 2.3.B؛

د) المكون **encryptedContentInfo** سيختزن حالة نمط البيانات **EncryptedContentInfo** على النحو المحدد في الفقرة 4.3.B؛

هـ) المكون **unprotectedAttrs** مطلوب إذا كان من المتوقع أن المكون التالي الذي سيُرسل في الاتجاه المعني، حالة نمط المحتوى **ct-authEnvelopedData**، وخلاف ذلك، يمكن أن يكون غائباً.

2.3.B معلومات المتلقي

يسمح نمط البيانات **RecipientInfos** بحالات متعددة من نمط البيانات **RecipientInfo**. ولكنه يجب أن يقتصر على حالة واحدة لغرض ملف التعريف هذا. وتوصِّف حالة نمط البيانات **RecipientInfo** خيارات مختلفة بشأن كيفية إنشاء السر المشترك بين طرفين يتواصلان.

```
RecipientInfos ::= SET SIZE (1) OF RecipientInfo

RecipientInfo ::= CHOICE {
  ktri      KeyTransRecipientInfo,
  kari  [1] KeyAgreeRecipientInfo,
  kekri [2] KEKRecipientInfo,
  pwri  [3] PasswordRecipientinfo,
  ori   [4] OtherRecipientInfo,
  ... }
```

ويستخدم ملف التعريف هذا اثنين من خيارات **RecipientInfo** على النحو المحدد في المرجع [IETF RFC 5652]:

أ) الخيار **kari** هو الخيار الوحيد الذي سيُستخدَم لنمط المحتوى **envelopedData**. ويقدم نمط البيانات **KeyAgreeRecipientInfo** المعلومات المطلوبة لإنشاء السر المشترك على النحو المفصل في الفقرة 3.3.B؛

ب) الخيار **kekri** هو الخيار الوحيد الذي سيُستخدَم لنمط المحتوى **ct-authEnvelopedData**. ويقدم نمط البيانات **KEKRecipientInfo** المعلومات المطلوبة لإنشاء السر المشترك على النحو المفصل في الفقرة 4.B.

3.3.B اتفاق المفتاح

```
KeyAgreeRecipientInfo ::= SEQUENCE {
  version                 CMSVersion (v3),
  originator          [0] EXPLICIT OriginatorIdentifierOrKey,
  ukm                 [1] EXPLICIT UserKeyingMaterial OPTIONAL,
  keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
  recipientEncryptedKeys RecipientEncryptedKeys,
  ... }

OriginatorIdentifierOrKey ::= CHOICE {
```

```
  issuerAndSerialNumber    IssuerAndSerialNumber,
  subjectKeyIdentifier [0] SubjectKeyIdentifier,
  originatorKey        [1] OriginatorPublicKey,
  ... }

OriginatorPublicKey ::= SEQUENCE {
  algorithm AlgorithmIdentifier {{SupportedDHPublicKeyAlgorithms}},
  publicKey BIT STRING,
  ... }

SupportedDHPublicKeyAlgorithms ALGORITHM ::= {...}

UserKeyingMaterial ::= OCTET STRING (SIZE (64))

KeyEncryptionAlgorithmIdentifier ::=
    AlgorithmIdentifier{{SupportedKeyIncryptAlgorithms}}

SupportedKeyIncryptAlgorithms ALGORITHM ::= {...}

RecipientEncryptedKeys ::= SEQUENCE (SIZE (1)) OF RecipientEncryptedKey

RecipientEncryptedKey ::= SEQUENCE {
  rid          KeyAgreeRecipientIdentifier,
  encryptedKey EncryptedKey }

KeyAgreeRecipientIdentifier ::= CHOICE {
  issuerAndSerialNumber IssuerAndSerialNumber,
--rKeyId             [0] IMPLICIT RecipientKeyIdentifier,
  ... }

EncryptedKey ::= OCTET STRING
```

لنمط البيانات **KeyAgreeRecipientInfo** المكونات التالية:

أ) مكون **النسخة** (**version**) سيتخذ القيمة **v3** وفقاً للمرجع [IETF RFC 5652]؛

ب) المكون **المنشئ** (**originator**) سيختزن حالة نمط البيانات **OriginatorIdentifierOrKey** مع الخيارات التالية:

– يتعين الأخذ بالخيار **issuerAndSerialNumber** إذا استُخدم الوضع الساكن– الساكن. ويتعين أن يختزن حالة نمط البيانات **IssuerAndSerialNumber**. ويتعين أن يحدد نمط البيانات هذا شهادة المفتاح العمومي DH للمرسل.

'1' ويتعين على مكون **المصدِر** (**issuer**) أن يحمل الاسم المميز لسلطة إصدار الشهادات (CA) المصدِرة، ويجب أن يكون مساوياً لمكون **مصدِر** (**issuer**) شهادة المفتاح العمومي المعنية؛

'2' ويجب أن يكون مكون **الرقم التسلسلي** (**serialNumber**) مساوياً لمكون الرقم التسلسلي لشهادة المفتاح العمومي المعنية؛

– ولا يجوز أن يؤخذ بالخيار **subjectKeyIdentifier**؛

– ويتعين الأخذ بالخيار **originatorKey** لوضع ديفي–هيلمان سريع الزوال – الساكن. ويتعين أن يختزن حالة نمط البيانات **OriginatorPublicKey** مع المكونان التاليان:

'1' مكون **الخوارزمية** (**algorithm**) الذي يجب أن يختزن إحالة إلى خوارزمية المفتاح العمومي ديفي–هيلمان المستخدمة؛

الملاحظة 1 – لا يفرض ملف التعريف هذا خوارزمية محددة للمفتاح العمومي ديفي–هيلمان. وتمكن الاستعاضة عن المواصفات المرجعية أو اتفاقات المنفذ بنقاط مع مجموعة من خوارزميات للمفتاح العمومي ديفي–هيلمان تُدعم في بيئة معينة.

'2' المكون **publicKey** الذي يجب أن يختزن المفتاح العمومي ديفي–هيلمان على النحو الذي ينشئه المرسل. ويتعين أن ينشئ المرسل زوجاً جديداً من مفاتيح ديفي–هيلمان لكل استخدام لنمط المحتوى هذا.

ويمكن للمرسل توليد السر المشترك من المفتاح الخاص المحلي والمفتاح العمومي للمتلقي. ويتعين أن ينشئ المتلقي سراً مشتركاً متطابقاً باستخدام مفتاحه الخاص إلى جانب المفتاح العمومي للمرسل كما يرد في نمط البيانات **OriginatorPublicKey** أو **IssuerAndSerialNumber**.

ج) المكون **ukm** الذي يجب أن يكون حاضراً عندما يُستخدم الوضع الساكن–الساكن وأن يختزن حالة نمط البيانات **UserKeyingMaterial**.

وينشئ كلا الطرفين ما يسمى بمفتاح تجفير المفتاح (KEK) من السر المشترك، والقيمة الواردة في المكون **ukm** (عند الاقتضاء)، وبعض المعلومات الأخرى على النحو المحدد في المرجع [IETF RFC 2631]. ويُستخدم هذا المفتاح لاحقاً لتجفير مفتاح تجفير المحتوى (CEK) الذي أنشأه المرسل. ويشار إلى هذه التقنية بتغليف المفتاح؛

د) المكون **keyEncryptionAlgorithm** الذي يجب أن يوصِّف خوارزمية تغليف المفتاح وأن يختزن حالة نمط البيانات **KeyEncryptionAlgorithmIdentifier**؛

الملاحظة 2 – لا يفرض ملف التعريف هذا مجموعة محددة من خوارزميات تغليف المفتاح. ويمكن تعريف خوارزميات جديدة في المستقبل. ويرد في المرجع [IETF RFC 3394] تعريف خوارزميات تغليف مفتاح معيار التجفير المتقدم (AES). وتمكن الاستعاضة عن المواصفات المرجعية أو اتفاقات المنفذ بنقاط مع مجموعة من خوارزميات التغليف تُدعم في بيئة معينة.

هـ) المكون **recipientEncryptedKeys** الذي يجب أن يختزن حالة نمط البيانات **RecipientEncryptedKeys**. ويتعين أن تتكون هذه الحالة من عنصر واحد فقط، أي حالة واحدة من نمط البيانات **RecipientEncryptedKey**. ولهذا النمط من البيانات المكونان التاليان:

– المكون **rid** الذي يجب أن يختزن هوية المتلقي التي حددتها شهادة المفتاح العمومي للكيان النهائي؛

– المكون **encryptedKey** الذي يجب أن يختزن مفتاح CEK المجفر المستخدم لتجفير المحتوى، كما نوقش في إطار الفقرة ج).

4.3.B إعادة استخدام مفاتيح تجفير محتوى قواعد تركيب الرسالة التجفيرية (CMS)

إذا أريدَ استخدام مفتاح CEK لحالة لاحقة من نمط المحتوى **ct-authEnvelopedData** على النحو المحدد في المرجع [IETF RFC 3185]، يتعين إدخال المعلومات المرجعية المناسبة في نعوت غير محمية كما نوقش في الفقرة 1.3.B. وعلى الطرفين الاحتفاظ بهذه المعلومات. وإذا لزم مستوى عال من الأمن، ينبغي أن يتخذ نعت من نمط **aa-CEKMaxDecrypts** قيمة '1' أو أن يُحذف.

5.3.B معلومات المحتوى المجفَّرة

تختزن حالة نمط البيانات **EncryptedContentInfo** المحتوى المغلف المجفَّر.

```
EncryptedContentInfo ::= SEQUENCE {
  contentType                 CONTENT-TYPE.&id ({EncryptedContentSet}),
  contentEncryptionAlgorithm SEQUENCE {
    algorithm                   ALGORITHM.&id ({SymmetricEncryptionAlgorithms}),
    parameter                   ALGORITHM.&Type
            ({SymmetricEncryptionAlgorithms}{@.algorithm})} OPTIONAL,
  encryptedContent       [0] IMPLICIT EncryptedContent OPTIONAL,
  ... }

EncryptedContentSet CONTENT-TYPE ::= {...}

SymmetricEncryptionAlgorithms ALGORITHM ::= {...}
```

```
EncryptedContent ::= OCTET STRING
```

ويتعين أن يختزن مكون نمط المحتوى **encryptedContentInfo** حالة نمط البيانات **EncryptedContentInfo**.

أ) يتعين أن يختزن المكون **contentType** نمط المحتوى للمحتوى المجفَّر. وتتألف قائمة أنماط المحتوى الممكنة من تلك التي يكون تجفيرها خياراً متاحاً؛

ب) ويرد المكونان **contentEncryptionAlgorithm** و**encryptedContent** على النحو المطلوب في المرجع [IETF RFC 5652].

4.B استخدام نمط محتوى البيانات المغلَّف المستيقَن منه

1.4.B اعتبارات عامة

على النحو المحدد في التوصية [ITU-T X.1080.1]، تتكون البروتوكولات العامة لبيانات القياس الحيوي عن بُعد من تبادل إعدادات لإقامة دورة تليه تبادلات متعددة للمعلومات تنتهي بانتهاء الدورة. وفي مثل هذه البيئة، قد لا يلزم إنشاء مفتاح تجفير مفتاح جديد لكل رسالة.

ويحدد المرجع [IETF RFC 5083] نمط المحتوى **ct-authEnvelopedData** غير المدرج في المرجع [IETF RFC 5652]. ويسمح نمط المحتوى هذا باستخدام تقنيات فعالة للتجفير المستيقَن منه. ويستخدم ملف التعريف هذا خوارزميات AES-GCM كما يرد تعريفها في المرجع [IETF RFC 5084] إلى جانب إعادة استخدام مفتاح CEK على النحو المحدد في المرجع [IETF RFC 3185]. وللاطلاع على التفاصيل، تنبغي مطالعة هذه المواصفات.

ويعرَّف نمط المحتوى **ct-authEnvelopedData** على النحو التالي:

```
ct-authEnvelopedData CONTENT-TYPE ::= {
                AuthEnvelopedData
  IDENTIFIED BY id-ct-authEnvelopedData }

AuthEnvelopedData ::= SEQUENCE {
  version                   CMSVersion (v0),
  originatorInfo       [0] IMPLICIT OriginatorInfo OPTIONAL,
  recipientInfos            RecipientInfos,
  authEncryptedContentInfo EncryptedContentInfo,
  authAttrs            [1] IMPLICIT Attributes {{AuthAttributes}} OPTIONAL,
  mac                       MessageAuthenticationCode,
  unauthAttrs          [2] IMPLICIT Attributes {{UnauthAttributes}} OPTIONAL }

AuthAttributes ATTRIBUTE ::= {...}

MessageAuthenticationCode ::= OCTET STRING

UnauthAttributes ATTRIBUTE ::=
  { aa-CEKReference | aa-CEKMaxDecrypts | aa-KEKDerivationAlg }
```

ولنمط البيانات **AuthEnvelopedData** المكونات التالية:

أ) مكون **النسخة** (**version**) سيتخذ القيمة **v0** وفقاً للمرجع [IETF RFC 5083]؛

ب) المكون **originatorInfo** يجب أن يكون غائباً؛

ج) المكون **recipientInfos** سيختزن حالة نمط البيانات **RecipientInfos**. ويرد وصف هذا النمط من البيانات في الفقرة 2.3.B. والخيار **kekri**، بالإضافة إلى الخيار **kari**، على صلة بنمط المحتوى هذا. وعندما يؤخذ بالخيار **kekri**، يتعين أن يختزن هذا الخيار حالة نمط البيانات **KEKRecipientInfo** على النحو المحدد في الفقرة 2.4.B؛

د) المكون **authEncryptedContentInfo** سيختزن حالة نمط البيانات **EncryptedContentInfo** على النحو المحدد في الفقرة 5.3.B؛

ه) المكون **authAttrs**، عندما يكون موجوداً، سيختزن مجموعة النعوت المزمع وضعها تحت حماية الاستيقان؛

و) المكون **mac** سيختزن شفرة الاستيقان من الرسالة (MAC) المتولدة؛

ز) المكون **unauthAttrs** يختزن نعوتاً من نفس النمط على النحو المحدد في الفقرة 1.3.B، البند ه). وإذا عُلم أن حالة نمط المحتوى هي الأخيرة في الدورة المعنية في اتجاه معين، يمكن عندئذ أن يكون هذا المكون غائباً.

2.4.B معلومات متلقي مفتاح تجفير المفتاح (KEK)

```
KEKRecipientInfo ::= SEQUENCE {
  version                CMSVersion (v4),
  kekid                  KEKIdentifier,
  keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
  encryptedKey           EncryptedKey }

KEKIdentifier ::= SEQUENCE {
  keyIdentifier OCTET STRING,
  date          GeneralizedTime OPTIONAL,
  other         OtherKeyAttribute OPTIONAL,
  ... }
```

لنمط البيانات **KEKRecipientInfo** المكونات التالية:

أ) مكون **النسخة** (**version**) سيتخذ القيمة **v4** وفقاً للمرجع [ETF RFC 5652]؛

ب) المكون **kekid** سيختزن حالة نمط البيانات **KEKIdentifier** مع المكونات التالية:

– المكون **KeyIdentifier** سيختزن معرف مفتاح CEK من مفتاح سابق جرى تبادله على النحو المحدد في الفقرة 4.3.B؛

– المكونات الأخرى ليست ضرورية؛

ج) المكون **keyEncryptionAlgorithm** سيختزن خوارزمية التغليف على النحو المحدد في الفقرة 3.3.B، البند د). ويوصى باستخدام نفس خوارزمية التغليف لجميع حالات المحتوى في دورة معينة لبيانات القياس الحيوي عن بُعد.

5.B النعوت

يرد في المرجع [IETF RFC 5652] تعريف نمط النعوت التالية. والمقصود من حالات أنماط النعت هذه أن تُدرَج كنعوت موقَّعة.

```
contentType ATTRIBUTE ::= {
  WITH SYNTAX            CONTENT-TYPE.&id({envelopedData, ...})
  EQUALITY MATCHING RULE objectIdentifierMatch
  SINGLE VALUE           TRUE
  ID                     id-contentType }

messageDigest ATTRIBUTE ::= {
  WITH SYNTAX            OCTET STRING
  EQUALITY MATCHING RULE octetStringMatch
  SINGLE VALUE           TRUE
  ID                     id-messageDigest }
```

يرد في المرجع [IETF RFC 3185] تعريف نمط النعوت التالية. ويمكن لحالات أنماط النعت هذه أن تُدرَج كنعوت غير موقَّعة.

```
aa-CEKReference ATTRIBUTE ::= {
  WITH SYNTAX            CEKReference
  EQUALITY MATCHING RULE octetStringMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-CEKReference }
```

```
CEKReference ::= OCTET STRING
aa-CEKMaxDecrypts ATTRIBUTE ::= {
  WITH SYNTAX            CEKMaxDecrypts
  EQUALITY MATCHING RULE integerMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-CEKMaxDecrypts }

CEKMaxDecrypts ::= INTEGER
aa-KEKDerivationAlg ATTRIBUTE ::= {
  WITH SYNTAX            KEKDerivationAlgorithm
  EQUALITY MATCHING RULE integerMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-KEKDerivationAlg }

KEKDerivationAlgorithm ::= SEQUENCE {
  kekAlg       AlgorithmIdentifier,
  pbkdf2Param  PBKDF2-params }
PBKDF2-params ::= SEQUENCE {
  salt CHOICE {
    specified   OCTET STRING,
--  otherSource AlgorithmIdentifier {{PBKDF2-SaltSources}}
    ... },
  iterationCount INTEGER (1..MAX),
  keyLength      INTEGER (1..MAX) OPTIONAL,
  prf            AlgorithmIdentifier {{PBKDF2-PRFs}},
  ... }

PBKDF2-PRFs ALGORITHM ::= {...}

PBKDF2-params ::= SEQUENCE {
  salt CHOICE {
    specified   OCTET STRING,
    otherSource AlgorithmIdentifier {{PBKDF2-SaltSources}} },
  iterationCount INTEGER (1..MAX),
  keyLength      INTEGER (1..MAX) OPTIONAL,
  prf            AlgorithmIdentifier {{PBKDF2-PRFs}} DEFAULT algid-hmacWithSHA1
}

id-pkcs OBJECT IDENTIFIER ::=
   { iso(1) member-body(2) usa(840) rsadsi(113549) pkcs(1) }

id-pkcs-9 OBJECT IDENTIFIER ::= { id-pkcs pkcs-9(9) }

id-aa OBJECT IDENTIFIER ::= { id-pkcs-9 smime(16) attributes(2) }

id-contentType         OBJECT IDENTIFIER ::= { id-pkcs-9 3 }
id-messageDigest       OBJECT IDENTIFIER ::= { id-pkcs-9 4 }
id-aa-CEKReference     OBJECT IDENTIFIER ::= { id aa 30 }
id-aa-CEKMaxDecrypts   OBJECT IDENTIFIER ::= { id aa 31 }
id-aa-KEKDerivationAlg OBJECT IDENTIFIER ::= { id aa 32 }
```

6.B شفرات الخطأ في قواعد تركيب الرسالة التجفيرية

ترد في المرجع [b-IETF RFC 7191] قائمة شفرات الخطأ في قواعد تركيب الرسالة التجفيرية (CMS) لجميع الاستخدامات الممكنة لقواعد تركيب الرسالة التجفيرية. وترد أدناه مجموعة فرعية من شفرات الخطأ هذه ذات الصلة ببيانات القياس الحيوي عن بُعد. وللاطلاع على وصف شفرات الخطأ، يرجى الرجوع إلى المرجع [b-IETF RFC 7191].

وعندما يشير المرجع [b-IETF RFC 7191] إلى شهادة، فهي إشارة إلى شهادة المفتاح العمومي المستخدمة في أنماط المحتوى المعرَّفة بقواعد تركيب الرسالة التجفيرية (CMS).

```
CmsErrorCode ::= ENUMERATED {
  decodeFailure                     (1),
  badContentInfo                    (2),
  badSignedData                     (3),
  badEncapContent                   (4),
  badCertificate                    (5),
  badSignerInfo                     (6),
  badSignedAttrs                    (7),
  badUnsignedAttrs                  (8),
  missingContent                    (9),
  noTrustAnchor                    (10),
  notAuthorized                    (11),
  badDigestAlgorithm               (12),
  badSignatureAlgorithm            (13),
  unsupportedKeySize               (14),
  unsupportedParameters            (15),
  signatureFailure                 (16),
  incorrectTarget                  (23),
  missingSignature                 (29),
  versionNumberMismatch            (31),
  revokedCertificate               (33),
  badEncryptedData                 (62),
  badEnvelopedData                 (63),
  badKeyAgreeRecipientInfo         (66),
  badKEKRecipientInfo              (67),
  badEncryptContent                (68),
  badEncryptAlgorithm              (69),
  missingCiphertext                (70),
  decryptFailure                   (71),
  badMACAlgorithm                  (72),
  badAuthAttrs                     (73),
  badUnauthAttrs                   (74),
  invalidMAC                       (75),
  mismatchedDigestAlg              (76),
  missingCertificate               (77),
  tooManySigners                   (78),
  missingSignedAttributes          (79),
  derEncodingNotUsed               (80),
  invalidAttributeLocation         (82),
  badAttributes                    (85),
  noMatchingRecipientInfo          (91),
  unsupportedKeyWrapAlgorithm      (92),
  badKeyTransRecipientInfo         (93),
  other                           (127) }
```

الملحق C

التوصيف الرسمي لبروتوكولات تأكيد الامتياز وتخصيصه

(يشكل هذا الملحق جزءاً أساسياً من هذه التوصية)

```
Pbact-access { joint-iso-itu-t(2) telebiometrics(42) e-health-protocol(3)
   modules(0) pbact-access(6) version1(1) }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- EXPORTS All

IMPORTS

  -- from Rec. ITU-T X.501 | ISO/IEC 9594-2

  ATTRIBUTE, Attribute{}, AttributeType, AttributeTypeAndValue,
  AttributeValueAssertion, DistinguishedName, OBJECT-CLASS, SupportedAttributes
    FROM InformationFramework {joint-iso-itu-t ds(5) module(1)
         informationFramework(1) 8}

  -- from Rec. ITU-T X.509 | ISO/IEC 9594-8

  AttributeCertificate
    FROM AttributeCertificateDefinitions {joint-iso-itu-t ds(5) module(1)
         attributeCertificateDefinitions(32) 8}

  CmsErrorCode, CONTENT-TYPE
    FROM CmsTelebiometric { joint-iso-itu-t(2) telebiometrics(42) th(3) part0(0)
         modules(0) cmsProfile(1) version1(1) } ;

accessService ATTRIBUTE ::= {
  WITH SYNTAX  AccessService
  ID           id-at-accessService }

AccessService ::= SEQUENCE {
  serviceId          OBJECT IDENTIFIER,
  objectDef          SEQUENCE SIZE (1..MAX) OF ObjectSel,
  ... }

ObjectSel ::= SEQUENCE {
  objecClass         OBJECT-CLASS.&id,
  objSelect          CHOICE {
    allObj        [0]  TargetSelect,
    objectNames   [1]  SEQUENCE SIZE (1..MAX) OF SEQUENCE {
      object             CHOICE {
        names         [1]  SEQUENCE SIZE (1..MAX) OF DistinguishedName,
        subtree       [2]  DistinguishedName,
        ... },
      select           TargetSelect,
      ... },
    ... },
  ... }

TargetSelect ::= SEQUENCE {
  objOper   ObjectOperations OPTIONAL,
  attrSel   AttributeSel     OPTIONAL,
  ... }
```

```
  (WITH COMPONENTS {..., objOper  PRESENT } |
   WITH COMPONENTS {..., attrSel  PRESENT } )


AttributeSel ::= SEQUENCE {
  attSelect           CHOICE {
    allAttr        [0]  SEQUENCE {
      attrOper1      [0]  AttributeOperations OPTIONAL,
      ... },
    attributes     [1]  SEQUENCE SIZE (1..MAX) OF SEQUENCE {
      select              SEQUENCE SIZE (1..MAX) OF ATTRIBUTE.&id,
      attrOper2      [0]  AttributeOperations OPTIONAL,
      ... },
    ... },
  ... }

ObjectOperations ::= BIT STRING {
  read            (0),
  add             (1),
  modify          (2),
  delete          (3),
  rename          (4),
  discloseOnError (5) }

AttributeOperations ::= BIT STRING {
  read             (0),
  compare          (1),
  add              (2),
  modify           (3),
  delete           (4),
  deleteValue      (5),
  replaceAttribute (6),
  discloseOnError  (7) }

PbactContentTypes CONTENT-TYPE ::= {
    privAssignRequest |
    privAssignResult |
    readRequest |
    readResult |
    compareRequest |
    compareResult |
    addRequest |
    addResult |
    deleteRequest |
    deleteResult |
    modifyRequest |
    modifyResult |
    renameRequest |
    renameResult,
    ... }

CommonReqComp ::= SEQUENCE {
  attrCerts [31]  AttributeCertificates OPTIONAL,
  serviceId [30]  OBJECT IDENTIFIER,
  invokId   [29]  INTEGER,
  ... }

AttributeCertificates ::= SEQUENCE SIZE (1..MAX) OF AttributeCertificate

readRequest CONTENT-TYPE ::= {
              ReadRequest
IDENTIFIED BY id-readRequest }

ReadRequest ::= SEQUENCE {
```

```
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  selection [2] InformationSelection,
  ... }

readResult CONTENT-TYPE ::= {
              ReadResult
IDENTIFIED BY id-readResult }


ReadResult ::= SEQUENCE {
  object    DistinguishedName,
  result    CHOICE {
    success   [0] ObjectInformation,
    failure   [1] AccessdErr,
    ... },
  ... }

compareRequest CONTENT-TYPE ::= {
              CompareRequest
IDENTIFIED BY id-compareRequest }

CompareRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  purported [2] AttributeValueAssertion,
  ... }

compareResult CONTENT-TYPE ::= {
              CompareResult
IDENTIFIED BY id-compareResult }

CompareResult ::= SEQUENCE {
  object    DistinguishedName,
  result    CHOICE {
    success   [0] CompareOK,
    failure   [1] AccessdErr,
    ... },
  ... }

CompareOK ::= SEQUENCE {
  matched        [0] BOOLEAN,
  matchedSubtype [1] BOOLEAN DEFAULT FALSE,
  ... }

addRequest CONTENT-TYPE ::= {
              AddRequest
IDENTIFIED BY id-addRequest }

AddRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object    [1] DistinguishedName,
  attr      [2] SEQUENCE SIZE (1..MAX) OF Attribute {{SupportedAttributes}}
                OPTIONAL,
  ... }

addResult CONTENT-TYPE ::= {
              AddResult
IDENTIFIED BY id-addResult }

AddResult ::= CHOICE {
  success   [0] NULL,
  failure   [1] AccessdErr,
  ... }
```

```
deleteRequest CONTENT-TYPE ::= {
              DeleteRequest
IDENTIFIED BY id-deleteRequest }

DeleteRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object        DistinguishedName,
  ... }

deleteResult CONTENT-TYPE ::= {
              DeleteResult
IDENTIFIED BY id-deleteResult }

DeleteResult ::= CHOICE {
  success   [0] NULL,
  failure   [1] AccessdErr,
  ... }

modifyRequest CONTENT-TYPE ::= {
              ModifyRequest
IDENTIFIED BY id-modifyRequest }

ModifyRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object        DistinguishedName,
  changes       SEQUENCE SIZE (1..MAX) OF ObjectModification,
  select        InformationSelection,
  ... }

ObjectModification ::= CHOICE {
  addAttribute     [0]  Attribute{{SupportedAttributes}},
  deleteAttribute  [1]  AttributeType,
  addValues        [2]  Attribute{{SupportedAttributes}},
  deleteValues     [3]  Attribute{{SupportedAttributes}},
  replaceAttribute [4]  Attribute{{SupportedAttributes}},
  ... }

modifyResult CONTENT-TYPE ::= {
              ModifyResult
IDENTIFIED BY id-modifyResult }

ModifyResult ::= SEQUENCE {
  result    CHOICE {
    success   [0] ObjectInformation,
    failure   [1] AccessdErr,
    ... },
  ... }

renameRequest CONTENT-TYPE ::= {
              RenameRequest
IDENTIFIED BY id-renameRequest }

RenameRequest ::= SEQUENCE {
  COMPONENTS OF CommonReqComp,
  object        DistinguishedName,
  new           DistinguishedName,
  ... }

renameResult CONTENT-TYPE ::= {
              RenameResult
IDENTIFIED BY id-renameResult }

RenameResult ::= SEQUENCE {
```

```
  result    CHOICE {
    success   [0] NULL,
    failure   [1] AccessdErr,
    ... },
  ... }

AccessdErr ::= CHOICE {
  cmsErr     [0] CmsErrorCode,
  pbactErr   [1] PbactErr,
  ... }

InformationSelection ::= SEQUENCE {
  attributes      CHOICE {
    allAttributes   [0]  NULL,
    select          [1]  SEQUENCE SIZE (1..MAX) OF ATTRIBUTE.&id,
    ... },
  infoTypes        ENUMERATED {
    attributeTypesOnly     (0),
    attributeTypeAndValue  (1),
    ... },
  ... }

ObjectInformation ::= SEQUENCE {
  name   DistinguishedName,
  info   SET SIZE (1..MAX) OF Attribute {{SupportedAttributes}},
  ... }

PbactErr ::= ENUMERATED {
  noSuchService,
  invalidOperationForService,
  insufficientAccessRight,
  noSuchObject,
  noSuchAttribute,
  noSuchAttributeValue,
  objectAlreadyExists,
  attributeAlreadyExists,
  attributeValueAlreadyExists,
  noInformation,
  ... }

privAssignRequest CONTENT-TYPE ::= {
              PrivAssignRequest
IDENTIFIED BY id-privAssignRequest }

PrivAssignRequest ::= SEQUENCE {
  attrCerts [1]  AttributeCertificates OPTIONAL,
  ... }

privAssignResult CONTENT-TYPE ::= {
              PrivAssignResult
IDENTIFIED BY id-privAssignResult }

PrivAssignResult ::= SEQUENCE {
  result  CHOICE {
    success  NULL,
    failure  PrivAssignErr },
  ... }

PrivAssignErr ::= CHOICE {
  cmsErr     [0] CmsErrorCode,
  assignErr  [1] AssignErr,
  ... }

AssignErr ::= ENUMERATED {
```

```
  invalidAttributeCertificate (0),
  ... }

-- object identifier allocations

-- top tree

id-pbact             OBJECT IDENTIFIER ::=
  { joint-iso-itu-t(2) telebiometrics(42) e-health-protocol(3) pbact(20) }
id-pbactmodule       OBJECT IDENTIFIER ::= { id-pbact module(0) }
id-pbactCont         OBJECT IDENTIFIER ::= { id-pbact cmsCont(1) }
id-pbactPrivAttr     OBJECT IDENTIFIER ::= { id-pbact prAttr(2) }

-- Content types

id-privAssignRequest OBJECT IDENTIFIER ::= { id-pbactCont privAssignRequest(1) }
id-privAssignResult  OBJECT IDENTIFIER ::= { id-pbactCont privAssignResult(2) }
id-readRequest       OBJECT IDENTIFIER ::= { id-pbactCont readRequest(3) }
id-readResult        OBJECT IDENTIFIER ::= { id-pbactCont readResult(4) }
id-compareRequest    OBJECT IDENTIFIER ::= { id-pbactCont compareRequest(5) }
id-compareResult     OBJECT IDENTIFIER ::= { id-pbactCont compareResult(6) }
id-addRequest        OBJECT IDENTIFIER ::= { id-pbactCont addRequest(7) }
id-addResult         OBJECT IDENTIFIER ::= { id-pbactCont addResult(8) }
id-deleteRequest     OBJECT IDENTIFIER ::= { id-pbactCont deleteRequest(9) }
id-deleteResult      OBJECT IDENTIFIER ::= { id-pbactCont deleteResult(10) }
id-modifyRequest     OBJECT IDENTIFIER ::= { id-pbactCont modifyRequest(11) }
id-modifyResult      OBJECT IDENTIFIER ::= { id-pbactCont modifyResult(12) }
id-renameRequest     OBJECT IDENTIFIER ::= { id-pbactCont renameRequest(13) }
id-renameResult      OBJECT IDENTIFIER ::= { id-pbactCont renameResult(14) }


-- Artribute types for carring privilege definitions

id-at-accessService OBJECT IDENTIFIER ::= { id-pbactPrivAttr 1 }

END
```

التذييل I

التوصيف غير الرسمي لملف تعريف قواعد تركيب الرسالة التجفيرية

(لا يشكل هذا التذييل جزءاً أساسياً من هذه التوصية)

إن التنفيذ الذي يقتصر على دعم وحدة **CmsTelebiometric** لا يلتزم بتوصيف فريق مهام هندسة الإنترنت (IETF) لقواعد تركيب الرسالة التجفيرية (CMS)، ويتعين ألا يُعتبر توصيفاً للتنفيذ. وهو يرد هنا لعرض المعلومات ولفحص الاتساق.

```
CmsTelebiometric { joint-iso-itu-t(2) telebiometrics(42) th(3) part0(0)
   modules(0) cmsProfile(1) version1(1) }
DEFINITIONS ::=
BEGIN

-- EXPORTS All

IMPORTS

  -- from Rec. ITU-T X.501 | ISO/IEC 9594-2

      ATTRIBUTE, Attribute{}, DistinguishedName, objectIdentifierMatch
    FROM InformationFramework {joint-iso-itu-t ds(5) module(1)
informationFramework(1) 8}

  -- from Rec. ITU-T X.509 | ISO/IEC 9594-8

  ALGORITHM, AlgorithmIdentifier, Certificate, CertificateSerialNumber
    FROM AuthenticationFramework {joint-iso-itu-t ds(5) module(1)
authenticationFramework(7) 8}

  -- from Rec. ITU-T X.520 | ISO/IEC 9594-6

  integerMatch, octetStringMatch
    FROM SelectedAttributeTypes {joint-iso-itu-t ds(5) module(1)
selectedAttributeTypes(5) 8}  ;

CONTENT-TYPE ::= TYPE-IDENTIFIER

ContentType ::= CONTENT-TYPE.&id

ContentInfo ::= SEQUENCE {
  contentType  CONTENT-TYPE.&id ({TelebSupportedcontentTypes}),
  content      CONTENT-TYPE.&Type
         ({TelebSupportedcontentTypes}{@contentType})OPTIONAL,
  ... }

TelebSupportedcontentTypes CONTENT-TYPE ::=
  { signedData | envelopedData | ct-authEnvelopedData, ...}

CMSVersion ::= INTEGER{ v0(0), v1(1), v2(2), v3(3), v4(4), v5(5) }

Attributes { ATTRIBUTE:AttrList } ::=
             SET SIZE (1..MAX) OF Attribute {{ AttrList }}

signedData CONTENT-TYPE ::= {
               SignedData
  IDENTIFIED BY id-signedData }

SignedData ::= SEQUENCE  {
```

```
  version               CMSVersion (v3),
  digestAlgorithms      SET (SIZE (1)) OF AlgorithmIdentifier
                          {{Teleb-Hash-Algorithms}},
  encapContentInfo      EncapsulatedContentInfo,
  certificates     [0]  IMPLICIT SET (SIZE (1..MAX)) OF Certificate OPTIONAL,
--crls             [1]  IMPLICIT RevocationInfoChoices OPTIONAL,
  signerInfos           SignerInfos,
  ... }

Teleb-Hash-Algorithms ALGORITHM ::= {...}

EncapsulatedContentInfo ::= SEQUENCE {
  eContentType     CONTENT-TYPE.&id({IncludedContent}),
  eContent     [0] EXPLICIT OCTET STRING
    (CONTAINING CONTENT-TYPE.&Type({IncludedContent}
     {@eContentType})) OPTIONAL }

IncludedContent CONTENT-TYPE ::= {envelopedData, ...}

SignerInfos ::= SET (SIZE (1)) OF SignerInfo

SignerInfo ::= SEQUENCE {
  version                 CMSVersion (v1),
  sid                     SignerIdentifier,
  digestAlgorithm         AlgorithmIdentifier {{Teleb-Hash-Algorithms}},
  signedAttrs        [0]  IMPLICIT Attributes{{SignedAttributes}} OPTIONAL,
  signatureAlgorithm      AlgorithmIdentifier {{Teleb-Signature-Algorithms}},
  signature               SignatureValue,
  unsignedAttrs      [1]  IMPLICIT Attributes {{UnsignedAttributes}} OPTIONAL,
  ... }

SignerIdentifier ::= CHOICE {
  issuerAndSerialNumber    IssuerAndSerialNumber,
--subjectKeyIdentifier [0] SubjectKeyIdentifier,
  ...}

IssuerAndSerialNumber ::= SEQUENCE {
  issuer       DistinguishedName,
  serialNumber CertificateSerialNumber }

SignedAttributes ATTRIBUTE ::= { contentType | messageDigest, ... }

Teleb-Signature-Algorithms ALGORITHM ::= {...}

SignatureValue ::= OCTET STRING

UnsignedAttributes ATTRIBUTE ::= {...}

envelopedData CONTENT-TYPE ::= {
                EnvelopedData
  IDENTIFIED BY id-envelopedData }

EnvelopedData ::= SEQUENCE {
  version                   CMSVersion(v0 | v2),
--originatorInfo         [0] IMPLICIT OriginatorInfo OPTIONAL,
  recipientInfos            RecipientInfos,
  encryptedContentInfo      EncryptedContentInfo,
  ...,
  [[2: unprotectedAttrs  [1] IMPLICIT Attributes
       {{UnprotectedAttributes}} OPTIONAL ]] }

RecipientInfos ::= SET SIZE (1) OF RecipientInfo

UnprotectedAttributes ATTRIBUTE ::=
```

```
  { aa-CEKReference | aa-CEKMaxDecrypts | aa-KEKDerivationAlg }
RecipientInfo ::= CHOICE {
--ktri      KeyTransRecipientInfo,
  kari  [1] KeyAgreeRecipientInfo,
  kekri [2] KEKRecipientInfo,
--pwri  [3] PasswordRecipientinfo,
--ori   [4] OtherRecipientInfo,
  ... }

KeyAgreeRecipientInfo ::= SEQUENCE {
  version                CMSVersion (v3),
  originator         [0] EXPLICIT OriginatorIdentifierOrKey,
  ukm                [1] EXPLICIT UserKeyingMaterial OPTIONAL,
  keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
  recipientEncryptedKeys RecipientEncryptedKeys,
  ... }

OriginatorIdentifierOrKey ::= CHOICE {
  issuerAndSerialNumber    IssuerAndSerialNumber,
--subjectKeyIdentifier [0] SubjectKeyIdentifier,
  originatorKey        [1] OriginatorPublicKey,
  ... }

OriginatorPublicKey ::= SEQUENCE {
  algorithm AlgorithmIdentifier {{SupportedDHPublicKeyAlgorithms}},
  publicKey BIT STRING,
  ... }

SupportedDHPublicKeyAlgorithms ALGORITHM ::= {...}

UserKeyingMaterial ::= OCTET STRING (SIZE (64))

KeyEncryptionAlgorithmIdentifier ::=
   AlgorithmIdentifier{{SupportedKeyIncryptAlgorithms}}

SupportedKeyIncryptAlgorithms ALGORITHM ::= {...}

RecipientEncryptedKeys ::= SEQUENCE (SIZE (1)) OF RecipientEncryptedKey
RecipientEncryptedKey ::= SEQUENCE {
  rid          KeyAgreeRecipientIdentifier,
  encryptedKey EncryptedKey }

KeyAgreeRecipientIdentifier ::= CHOICE {
  issuerAndSerialNumber IssuerAndSerialNumber,
--rKeyId            [0] IMPLICIT RecipientKeyIdentifier,
  ... }

EncryptedKey ::= OCTET STRING

EncryptedContentInfo ::= SEQUENCE {
  contentType                CONTENT-TYPE.&id ({EncryptedContentSet}),
  contentEncryptionAlgorithm SEQUENCE {
    algorithm                  ALGORITHM.&id ({SymmetricEncryptionAlgorithms}),
    parameter                  ALGORITHM.&Type
            ({SymmetricEncryptionAlgorithms}{@.algorithm})} OPTIONAL,
  encryptedContent       [0] IMPLICIT EncryptedContent OPTIONAL,
  ... }

EncryptedContentSet CONTENT-TYPE ::= {...}

SymmetricEncryptionAlgorithms ALGORITHM ::= {...}

EncryptedContent ::= OCTET STRING
```

```
ct-authEnvelopedData CONTENT-TYPE ::= {
                AuthEnvelopedData
  IDENTIFIED BY id-ct-authEnvelopedData }

AuthEnvelopedData ::= SEQUENCE {
  version                   CMSVersion (v0),
--originatorInfo       [0] IMPLICIT OriginatorInfo OPTIONAL,
  recipientInfos            RecipientInfos,
  authEncryptedContentInfo EncryptedContentInfo,
  authAttrs             [1] IMPLICIT Attributes {{AuthAttributes}} OPTIONAL,
  mac                       MessageAuthenticationCode,
  unauthAttrs           [2] IMPLICIT Attributes {{UnauthAttributes}} OPTIONAL }

AuthAttributes ATTRIBUTE ::= {...}

MessageAuthenticationCode ::= OCTET STRING

UnauthAttributes ATTRIBUTE ::=
  { aa-CEKReference | aa-CEKMaxDecrypts | aa-KEKDerivationAlg }

KEKRecipientInfo ::= SEQUENCE {
  version                CMSVersion (v4),
  kekid                  KEKIdentifier,
  keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
  encryptedKey           EncryptedKey }

KEKIdentifier ::= SEQUENCE {
  keyIdentifier OCTET STRING,
--date          GeneralizedTime OPTIONAL,
--other         OtherKeyAttribute OPTIONAL,
  ... }

contentType ATTRIBUTE ::= {
  WITH SYNTAX            CONTENT-TYPE.&id({envelopedData, ...})
  EQUALITY MATCHING RULE objectIdentifierMatch
  SINGLE VALUE           TRUE
  ID                     id-contentType }

messageDigest ATTRIBUTE ::= {
  WITH SYNTAX            OCTET STRING
  EQUALITY MATCHING RULE octetStringMatch
  SINGLE VALUE           TRUE
  ID                     id-messageDigest }

aa-CEKReference ATTRIBUTE ::= {
  WITH SYNTAX            CEKReference
  EQUALITY MATCHING RULE octetStringMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-CEKReference }

CEKReference ::= OCTET STRING
aa-CEKMaxDecrypts ATTRIBUTE ::= {
  WITH SYNTAX            CEKMaxDecrypts
  EQUALITY MATCHING RULE integerMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-CEKReference }

CEKMaxDecrypts ::= INTEGER

aa-KEKDerivationAlg ATTRIBUTE ::= {
  WITH SYNTAX            KEKDerivationAlgorithm
  EQUALITY MATCHING RULE integerMatch
  SINGLE VALUE           TRUE
  ID                     id-aa-KEKDerivationAlg }
```

```
KEKDerivationAlgorithm ::= SEQUENCE {
  kekAlg       AlgorithmIdentifier {{SupportedKeyIncryptAlgorithms}},
  pbkdf2Param  PBKDF2-params }

PBKDF2-params ::= SEQUENCE {
  salt CHOICE {
    specified   OCTET STRING,
--  otherSource AlgorithmIdentifier {{PBKDF2-SaltSources}}
    ... },
  iterationCount INTEGER (1..MAX),
  keyLength      INTEGER (1..MAX) OPTIONAL,
  prf            AlgorithmIdentifier {{PBKDF2-PRFs}},
  ... }

PBKDF2-PRFs ALGORITHM ::= {...}

id-pkcs OBJECT IDENTIFIER ::=
   { iso(1) member-body(2) usa(840) rsadsi(113549) pkcs(1) }

id-pkcs-9 OBJECT IDENTIFIER ::= { id-pkcs pkcs-9(9) }

id-ct OBJECT IDENTIFIER ::= { id-pkcs-9 smime(16) ct(1) }
id-aa OBJECT IDENTIFIER ::= { id-pkcs-9 smime(16) attributes(2) }

id-contentType         OBJECT IDENTIFIER ::= { id-pkcs-9 3 }
id-messageDigest       OBJECT IDENTIFIER ::= { id-pkcs-9 4 }
id-aa-CEKReference     OBJECT IDENTIFIER ::= { id-aa 30 }
id-aa-CEKMaxDecrypts   OBJECT IDENTIFIER ::= { id-aa 31 }
id-aa-KEKDerivationAlg OBJECT IDENTIFIER ::= { id-aa 32 }

id-signedData OBJECT IDENTIFIER ::= {iso(1) member-body(2)
us(840)rsadsi(113549) pkcs(1) pkcs7(7) 2}

id-envelopedData OBJECT IDENTIFIER ::= {iso(1) member-body(2) us(840)
rsadsi(113549) pkcs(1) pkcs7(7) 3}

id-ct-authEnvelopedData OBJECT IDENTIFIER ::= { id-ct 23 }

END -- CmsTelebiometric
```

بيبليوغرافيا

[b-ITU-T X.841] Recommendation ITU-T X.841 (2000) | ISO/IEC 15816:2002, *Information technology – Security techniques – Security information objects for access control*.

[b-IEC 62351-8] IEC TS 62351-8:2011, *Power systems management and associated information exchange– Data and communications security – Part 8: Role-based access control*.

[b-NIST 800-56A] NIST Special Publication 800-56A, Revision 2 (2013), *Recommendation for Pair-Wise Key-Establishment. Schemes Using Discrete Logarithm Cryptography*.

[b-NIST 800-162] NIST Special Publication 800-162 (2014), *Guide to Attribute Based Access Control (ABAC) Definition and Considerations*.

[b-IETF RFC 5480] IETF RFC 5480 (2009), *Elliptic Curve Cryptography Subject Public Key Information*.

[b-IETF RFC 7191] IETF RFC 7191 (2014), *Cryptographic Message Syntax (CMS) – Key Package Receipt and Error Content Types*.

سلاسل التوصيات الصادرة عن قطاع تقييس الاتصالات

السلسلة A	تنظيم العمل في قطاع تقييس الاتصالات
السلسلة D	مبادئ التعريفة والمحاسبة والقضايا الاقتصادية والسياساتية المتصلة بالاتصالات/تكنولوجيا المعلومات والاتصالات على الصعيد الدولي
السلسلة E	التشغيل العام للشبكة والخدمة الهاتفية وتشغيل الخدمات والعوامل البشرية
السلسلة F	خدمات الاتصالات غير الهاتفية
السلسلة G	أنظمة الإرسال ووسائطه والأنظمة والشبكات الرقمية
السلسلة H	الأنظمة السمعية المرئية والأنظمة متعددة الوسائط
السلسلة I	الشبكة الرقمية متكاملة الخدمات
السلسلة J	الشبكات الكبلية وإرسال إشارات تلفزيونية وبرامج صوتية وإشارات أخرى متعددة الوسائط
السلسلة K	الحماية من التداخلات
السلسلة L	البيئة وتكنولوجيا المعلومات والاتصالات، وتغير المناخ، والمخلفات الإلكترونية، وكفاءة استخدام الطاقة، وإنشاء الكبلات وغيرها من عناصر المنشآت الخارجية وتركيبها وحمايتها
السلسلة M	إدارة الاتصالات بما في ذلك شبكة إدارة الاتصالات وصيانة الشبكات
السلسلة N	الصيانة: الدارات الدولية لإرسال البرامج الإذاعية الصوتية والتلفزيونية
السلسلة O	مواصفات تجهيزات القياس
السلسلة P	نوعية الإرسال الهاتفي والمنشآت الهاتفية وشبكات الخطوط المحلية
السلسلة Q	التبديل والتشوير، والقياسات والاختبارات المرتبطة بهما
السلسلة R	الإرسال البرقي
السلسلة S	التجهيزات المطرافية للخدمات البرقية
السلسلة T	المطاريف الخاصة بالخدمات التليماتية
السلسلة U	التبديل البرقي
السلسلة V	اتصالات البيانات على الشبكة الهاتفية
السلسلة X	**شبكات البيانات والاتصالات بين الأنظمة المفتوحة ومسائل الأمن**
السلسلة Y	البنية التحتية العالمية للمعلومات، والجوانب الخاصة ببروتوكول الإنترنت وشبكات الجيل التالي وإنترنت الأشياء والمدن الذكية
السلسلة Z	اللغات والجوانب العامة للبرمجيات في أنظمة الاتصالات

طُبعت في سويسرا
جنيف، 2018