الاتحـــــاد الـــدولـــــي للاتصـــــــالات

ITU-T X.518

(2005/08)

قطـــاع تقييس الاتصـــالات
في الاتحاد الدولي للاتصالات

السلسلة X: شبكات المعطيات والاتصالات بين الأنظمة المفتوحة والأمن

الدليل

تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: إجراءات العمليات الموزعة

التوصيـــة ITU-T X.518

الاتحاد الدولي للاتصالات

توصيات السلسلة X الصادرة عن قطاع تقييس الاتصالات

شبكات المعطيات والاتصالات بين الأنظمة المفتوحة ومسائل الأمن

الشبكات العمومية للمعطيات	
الخدمات والمرافق	X.19 – X.1
السطوح البينية	X.49 – X.20
الإرسال والتشوير والتبديل	X.89 – X.50
جوانب الشبكة	X.149 – X.90
الصيانة	X.179 – X.150
الترتيبات الإدارية	X.199 – X.180
التوصيل البيني للأنظمة المفتوحة	
النموذج والترميز	X.209 – X.200
تعاريف الخدمات	X.219 – X.210
مواصفات البروتوكول بأسلوب التوصيل	X.229 – X.220
مواصفات البروتوكول بأسلوب غياب التوصيل	X.239 – X.230
جداول إعلان المطابقة (PICS)	X.259 – X.240
تعرّف هوية البروتوكول	X.269 – X.260
بروتوكولات الأمن	X.279 – X.270
أشياء مسيرة على الطبقة	X.289 – X.280
اختبار المطابقة	X.299 – X.290
التشغيل البيني للشبكات	
اعتبارات عامة	X.349 – X.300
الأنظمة الساتلية لإرسال البيانات	X.369 – X.350
الشبكات القائمة على بروتوكول الإنترنت	X.399 – X.370
أنظمة معالجة الرسائل	X.499 – X.400
الدليل	X.599 – X.500
التوصيل الشبكي في التوصيل البيني للأنظمة المفتوحة (OSI) وجوانب النظام	
التوصيل الشبكي	X.629 – X.600
الفعالية	X.639 – X.630
نوعية الخدمة	X.649 – X.640
التسمية والعنونة والتسجيل	X.679 – X.650
ترميز النظم المجرد واحد (ASN.1)	X.699 – X.680
إدارة التوصيل البيني للأنظمة المفتوحة (OSI)	
الإطار والهيكل المعماري لإدارة الأنظمة	X.709 – X.700
خدمة اتصالات الإدارة وبروتوكولاتها	X.719 – X.710
هيكل معلومات الإدارة	X.729 – X.720
وظائف الإدارة ووظائف الهيكل المعماري للإدارة الموزعة المفتوحة	X.799 – X.730
الأمن	X.849 – X.800
تطبيقات التوصيل البيني للأنظمة المفتوحة (OSI)	
الالتزام والتلازم والاستعادة	X.859 – X.850
معالجة المعاملات	X.879 – X.860
العمليات البُعدية	X.889 – X.880
التطبيقات التنوعية لترميز النظم المجرد واحد (ASN.1)	X.899 – X.890
المعالجة الموزعة المفتوحة	X.999 – X.900
أمن الاتصالات	X.1999 – X.1000

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

المعيار الدولي ISO/IEC 9594-4
التوصية ITU-T X.518

تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: إجراءات العمليات الموزعة

موجز

تعرف هذه التوصية | المعيار الدولي الإجراءات التي بموجبها توفر المكونات الموزعة للتشغيل البيني للدليل خدمة متسقة لمستعمليه.

المصدر

وافقت لجنة الدراسات 17 (2005-2008) لقطاع تقييس الاتصالات بتاريخ 29 أغسطس 2005 على التوصية ITU-T X.518 بموجب الإجراء المحدد في التوصية A.8. ونشر نص مطابق أيضاً باعتباره معيار ISO/IEC 9594-4.

تمهيـــد

الاتحاد الدولي للاتصالات وكالة متخصصة للأمم المتحدة في ميدان الاتصالات. 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 رقم 1 الصادر عن الجمعية العالمية لتقييس الاتصالات.

وفي بعض مجالات تكنولوجيا المعلومات التي تقع ضمن اختصاص قطاع تقييس الاتصالات، تعد 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 المعطيات الخاصة ببراءات الاختراع في مكتب تقييس الاتصالات (TSB) في الموقع http://www.itu.int/ITU-T/ipr/.

© ITU 2006

جميع الحقوق محفوظة. لا يجوز استنساخ أي جزء من هذه المنشورة بأي وسيلة كانت إلا بإذن خطي مسبق من الاتحاد الدولي للاتصالات.

جدول المحتويات

الصفحة

1 النطاق 1
2 المراجع المعيارية 1
1.2 التوصيات/المعايير الدولية المتطابقة 1
2.2 مراجع أخرى 2
3 تعاريف 2
1.3 تعاريف نموذج اتصالات 2
2.3 تعاريف الدليل الأساسية 2
3.3 تعاريف نموذج الدليل 2
4.3 تعاريف نموذج معلومات وكيل نظام الدليل 3
5.3 تعاريف خدمة مجردة 4
6.3 تعاريف استنساخ دليل 4
7.3 تعاريف عمليات موزعة 4
4 مختصرات 6
5 الاصطلاحات 6
6 نظرة شاملة 8
7 نموذج نظام دليل موزع 9
8 نموذج تفاعلات DSA 9
1.8 تجزئة طلب 10
2.8 سلسلة وحيدة 10
3.8 سلسلة متعددة 11
4.8 مرجع 12
5.8 تحديد الأسلوب 13
9 نظرة شاملة على الخدمة المجردة لـــ DSA 14
10 أنماط المعلومات 14
1.10 مقدمة 14
2.10 أنماط المعلومات المعرفة في أماكن أخرى 14
3.10 متغيرات سلسلة 15
4.10 نتائج سلسلة 19
5.10 تقدم عملية 19
6.10 معلومات أثر 20
7.10 نمط مرجع 20
8.10 معلومات نقطة نفاذ 21
9.10 معرفة جسر DIT 22
10.10 الاستبعاد 22
11.10 مراجع الإستمرارية 23
11 الربط وفك الربط 24
1.11 ربط DSA 24

2.11 فك ربط DSA 25
12 عمليات متسلسلة 25
1.12 عمليات متسلسلة 25
2.12 عملية ترك متسلسلة 26
3.12 عمليات متسلسلة وصيغة بروتوكول 26
13 أخطاء متسلسلة 27
1.13 مقدمة 27
2.13 مرجع DSA 27
14 مقدمة 28
1.14 المدى والحدود 28
2.14 المطابقة 28
3.14 نموذج مفهومي 28
4.14 عملية فردية وتعاونية لـــ DSAs 28
5.14 الاتفاقات التعاونية بين DSAs 29
15 سلوك دليل موزع 29
1.15 التنفيذ التعاوني لعمليات 29
2.15 مراحل معالجة عملية 29
3.15 إدارة عمليات موزعة 31
4.15 مناولة العروات 31
5.15 اعتبارات أخرى لعملية موزعة 32
6.15 استيقان عمليات موزعة 34
16 محول العمليات 35
1.16 مفاهيم عامة 35
2.16 إجراءات محول عمليات 40
3.16 نظرة شاملة على الإجراءات 41
17 إجراء اقرار صلاحية طلب 43
1.17 مقدمة 43
2.17 معلمات إجراء 44
3.17 تعريف إجراء 45
18 إجراء Name Resolution 47
1.18 مقدمة 47
2.18 معلمات إجراء Find DSE 48
3.18 إجراءات 49
19 تقييم عملية 61
1.19 إجراء تعديل 61
2.19 إجراء استفسار مدخل وحيد 70
3.19 إجراء استفسار مدخل متعدد 70
20 إجراءات مرجع استمرارية 86
1.20 استراتيجية سلسلة في وجود تظليل 87
2.20 اصدار طلبات فرعية متسلسلة إلى DSA عن بعد 89

3.20 معلمات إجراءات 89
4.20 تعريف الإجراءات 90
5.20 إجراء Abandon 99
21 إجراء Results Merging 100
22 إجراءات استيقان موزع 103
1.22 استيقان مرسل 103
2.22 استيقان نتائج 104
23 نظرة شاملة على إدارة معرفة 105
1.23 صيانة مراجع معرفة 105
2.23 طلب إسناد مرجعي 107
3.23 عدم اتساق معرفة 107
4.23 مراجع معرفة وسياقات 108
24 إسنادات تشغيلية تراتبية 109
1.24 خاصيات نمط إسناد تشغيلي 109
2.24 تعريف صنف شيء لمعلومات إسناد تشغيلي 112
3.24 إجراءات DSA لإدارة إسناد تشغيلي تراتبي 113
4.24 إجراءات عمليات 117
5.24 استخدام سياقات تطبيق 117
25 إسناد تشغيلي تراتبي غير محدد 117
1.25 خاصيات نمط إسناد تشغيلي 118
2.25 تعريف صنف شيء لمعلومات إسناد تشغيلي 119
3.25 إجراءات DSA لإدارة إسناد تشغيلي تراتبي غير محدد 120
4.25 إجراءات للعمليات 122
5.25 استخدام سياقات تطبيق 122
الملحق A - ASN.1 لعمليات موزعة 123
الملحق B - مثال لاستبانة اسم موزع 127
الملحق C - الاستخدام الموزع للاستيقان 129
1.C موجز 129
2.C نموذج حماية موزعة 129
3.C عمليات متسلسلة موقعة 130
4.C عمليات متسلسلة مجفرة 131
5.C عمليات موزعة موقعة ومجفرة 134
الملحق D - مواصفة أنماط إسناد تراتبي وتشغيلي تراتبي غير محدد 136
الملحق E - مثال صيانة معرفة 139
الملحق F - التعديلات والتصويبات 142

مقدمة

وضعت هذه التوصية | المعيار الدولي، وكذلك التوصيات الأخرى/المعايير الدولية الأخرى، من أجل تيسير التوصيل البيني لأنظمة معالجة معلومات لتوفير خدمات الدليل. ويمكن أن تعتبر مجموعة هذه الأنظمة، وكذلك معلومات الدليل التي تحتويها، كُلاً متكاملاً، يسمى *الدليل*. وتستخدم المعلومات التي يحتويها الدليل، والمسماة جماعياً قاعدة معلومات الدليل (DIB)، بصفة عامة، لتيسير الاتصال بين الأشياء التي تعتبر كيانات تطبيق والأشخاص والمطاريف وقوائم التوزيع، أو الاتصال مع هذه الأشياء أو بشأنها.

يقوم الدليل بدور مهم في التوصيل البيني لأنظمة مفتوحة، غايته هي أن يمكن، بفضل حد أدنى من الاتفاقات التقنية خارج معايير التوصيل البيني نفسها، من تحقيق التوصيل البيني لأنظمة معالجة المعلومات:

- من مختلف المصنّعين؛
- التي تدار بطرق مختلفة؛
- التي هي على مستويات مختلفة من التعقيد؛
- التي هي ذات أعمار مختلفة.

تعرف هذه التوصية | المعيار الدولي الإجراءات التي بموجبها توفر المكونات الموزعة للتشغيل البيني للدليل خدمة متسقة لمستعمليه.

توفر هذه التوصية | المعيار الدولي أطر أساس يمكن بناءً عليها تعريف المظاهر الجانبية للصناعة من قبل مجموعات معيارية ومنتديات الصناعة. وكثير من الخاصيات المعرفة باعتبارها خيارية في هذه الأطر، يمكن أن تكون إلزامية لاستخدامها في بعض البيئات من خلال مظاهر جانبية. وتنقح الطبعة الخامسة هذه وتعزز تقنياً، ولكنها لا تحل محل الطبعة الرابعة لهذه التوصية | المعيار الدولي. ويمكن أن يطالب التنفيذ مطابقة الطبعة الرابعة. ومع ذلك، عند نقطة معينة، لا تدعم الطبعة الرابعة (أي، لن تحل العيوب التي تم الإبلاغ عنها). ويوصي بأن يتطابق التنفيذ مع الطبعة الخامسة هذه في أسرع وقت ممكن.

وتصف الطبعة الخامسة هاتين الصيغتين 1 و2 من بروتوكولات الدليل.

أما الطبعتان الأولى والثانية فتحددان الإصدار الأول فقط. ومعظم الخدمات والبروتوكولات المذكورة في هذه الطبعة مصممة للعمل بموجب الإصدار الأول، إلا أن بعض الخدمات والبروتوكولات المحسنة، مثل الأخطاء الموقعة، لا تعمل ما لم تشتمل جميع كيانات الدليل الداخلة في العملية على الإصدار 2 المتفاوض عليه. وأياً كان الإصدار المتفاوض عليه فإن هذه الطبعة تتيح التعامل مع الاختلافات بين الخدمات وبين البروتوكولات المحددة في الإصدارات الخمسة فيما عدا الخدمات والبروتوكولات المخصصة على وجه التحديد للإصدار 2، وذلك باستخدام قواعد قابلية التمديد الوارد تعريفها في ITU-T X.519 | ISO/IEC 9594-5.

يوفر الملحق A، الذي يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، وحدة ASN.1 للعمليات الموزعة للدليل.

يورد الملحق B، الذي لا يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، مثالاً لاستبانة اسم موزع.

يورد الملحق C، الذي لا يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، استيقان في بيئة عمليات موزعة.

يوفر الملحق D، الذي يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، تعاريف لأصناف شيء معلومات ASN.1 مقدمة في مواصفة هذا الدليل.

يوضح الملحق E، الذي لا يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، صيانة المعرفة.

يورد الملحق F، الذي لا يشكل جزءاً لا يتجزأ من هذه التوصية | المعيار الدولي، التعديلات وتقارير العيوب التي تم إدراجها لتشكل هذه الطبعة من هذه التوصية | المعيار الدولي.

المعيار الدولي
توصية قطاع تقييس الاتصالات للاتحاد الدولي للاتصالات

تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: إجراءات العمليات الموزعة

القسم 1 – عام

1 مجال التطبيق

تعرف هذه التوصية | المعيار الدولي سلوك DSAs المشاركين في تطبيق الدليل الموزع. وقد صُمم السلوك المسموح به لضمان خدمة متسقة لتوزيع واسع لـــ DIB عبر DSAs كثيرين.

ليس القصد من الدليل أن يكون نظاماً لقاعدة معطيات لغرض عام، بالرغم من امكانية بنائه على ذلك النظام. ويفترض وجود تردد عالي كبير للاستفسارات أكثر من التحيينات.

2 المراجع المعيارية

تتضمن التوصيات التالية لقطاع تقييس الاتصالات وغيرها من المراجع أحكاماً تشكل من خلال الإشارة إليها في هذا النص جزءاً لا يتجزأ 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.

1.2 التوصيات/المعايير الدولية المتطابقة

- التوصية ITU-T X.200 (1994) | ISO/IEC 7498-1:1994، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – النموذج المرجعي الأساسي: النموذج الأساسي.*
- ITU-T Recommendation X.500 (2005) | ISO/IEC 9594-1:2005, Information technology – *Open Systems Interconnection – The Directory: Overview of concepts, models and services.*
- التوصية ITU-T X.501 (2005) | ISO/IEC 9594-2:2005، تكنولوجيا المعلومات – *التوصيل البيني للأنظمة المفتوحة – الدليل: النماذج.*
- التوصية ITU-T X.509 (2005) | ISO/IEC 9594-8:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: أطر التصديق العمومية الرئيسية وتصديق النعوت، بالإضافة إلى التصويب التقني 1، والتصويب التقني 2 (2002)، والتصويب التقني 3 (2003).*
- التوصية ITU-T X.511 (2005) | ISO/IEC 9594-3:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: تعريف الخدمة المجردة.*
- التوصية ITU-T X.519 (2005) | ISO/IEC 9594-5:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: مواصفات البروتوكول.*

– التوصية ITU-T X.520 (2005) | ISO/IEC 9594-6:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: أنماط النعوت المنتقاة.*

– التوصية ITU-T X.521 (2005) | ISO/IEC 9594-7:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: فئات الموضوعات المنتقاة.*

– التوصية ITU-T X.525 (2005) | ISO/IEC 9594-9:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: النسخ.*

– التوصية ITU-T X.530 (2005) | ISO/IEC 9594-10:2005، *تكنولوجيا المعلومات – التوصيل البيني للأنظمة المفتوحة – الدليل: استخدام أساليب إدارة الأنظمة في إدارة الدليل.*

– ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002, Information technology – *Abstract Syntax Notation One (ASN.1): Specification of basic notation.*

– ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002, Information technology – *Abstract Syntax Notation One (ASN.1): Information object specification.*

– ITU-T Recommendation X.682 (2002) | ISO/IEC 8824-3:2002, Information technology – *Abstract Syntax Notation One (ASN.1): Constraint specification.*

– ITU-T Recommendation X.683 (2002) | ISO/IEC 8824-4:2002, Information technology – *Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications.*

2.2 مراجع أخرى

– IETF RFC 2251 (1997), *Lightweight Directory Access Protocol (v3).*

– IETF RFC 3377 (2002), *Lightweight Directory Access Protocol (v3): Technical Specification.*

3 تعاريف

لأغراض هذه التوصية | المعيار الدولي، تنطبق التعاريف التالية:

1.3 تعاريف نموذج اتصالات

التعاريف التالية محددة في التوصية ITU-T X.519 | ISO/IEC 9594-5:

أ) *عنوان كيان تطبيق*

2.3 تعاريف الدليل الأساسية

التعاريف التالية محددة في التوصية ITU-T X.500 | ISO/IEC 9594-1:

أ) *الدليل؛*

ب) *قاعدة معلومات الدليل.*

3.3 تعاريف نموذج الدليل

التعاريف التالية محددة في التوصية ITU-T X.501/ISO/IEC 9594-2:

أ) *نقطة نفاذ؛*

ب) *مستعار؛*

ج) *اسم مميز؛*

د) *شجرة معلومات الدليل؛*

هـ) *وكيل نظام الدليل (DSA)؛*

و) *وكيل مستعمل الدليل (DUA)؛*

ز) *اسم مميز نسبي.*

4.3 تعاريف نموذج معلومات وكيل نظام الدليل

التعاريف التالية محددة في التوصية ITU-T X.501/ISO/IEC 9594-2:

أ) *فئة؛*

ب) *قابل للاستخدام على نحو مشترك؛*

ج) *سابقة سياق؛*

د) *إسناد مرجعي؛*

هـ) *جزء قاعدة معلومات الدليل؛*

و) *شجرة معلومات وكيل نظام الدليل؛*

ز) *مدخل محدد – وكيل نظام الدليل؛*

ح) *نمط مدخل محدد لوكيل نظام الدليل؛*

ط) *مرجع رئيسي مباشر؛*

ي) *معلومات معرفة؛*

ك) *فئة مرجع معرفة؛*

ل) *نمط مرجع معرفة؛*

م) *سياق تسمية؛*

ن) *معرفة غير محددة؛*

س) *مرجع تابع غير محدد؛*

ع) *نعت تشغيلي؛*

ف) *مسير مرجع؛*

ص) *معرفة محددة؛*

ق) *مرجع تابع؛*

ر) *مرجع رئيسي.*

5.3 تعاريف خدمة مجردة

التعاريف التالية محددة في التوصية ITU-T X.511/ISO/IEC 9594-3:

أ) *نتيجة متدفقة.*

6.3 تعاريف استنساخ دليل

التعاريف التالية محددة في التوصية ITU-T X.525/ISO/IEC 9594-9:

أ) *إكمال نعت؛*

ب) *إسناد تشغيلي لتظليل؛*

ج) *إكمال تابع؛*

د) *وحدة استنساخ.*

7.3 تعاريف عمليات موزعة

التعاريف التالية محددة في هذه التوصية | المعيار الدولي:

1.7.3 شيء قاعدة: الشيء أو مدخل مستعار هو هدف لعملية كما اصدرها المرسل.

2.7.3 وكيل نظام دليل موثق: وكيل نظام الدليل (DSA) الذي يطلب من وكيل مستعمل الدليل (DUA) الموثق بواسطة أداء عملية إسناد مع DSA ذلك.

3.7.3 نتائج متصفحة لوكيل نظام دليل موثق: يقوم بأداء التصفح الكامل DSA الذي يوثق DUA.

ملاحظة – هذا هو الأسلوب الوحيد للتصفح الذي تدعمه الأنظمة المطابقة للطبعات قبل الطبعة الخامسة.

4.7.3 سلسلة: المصطلح التنوعي لسلسلة وحيدة أو سلسلة متعددة.

5.7.3 معلومات سابقة سياق: معلومات تشغيلية ولمستعمل يوردها DSA رئيسي إلى DSA تابع في RHOB يتعلق بـــذروات شجرة معلومات الدليل أعلى من سابقة سياق تابع.

6.7.3 استبانة اسم موزع: العملية التي يؤدى باسمها استبانة في أكثر من DSA واحد.

7.7.3 نتائج متصفحة لبروتوكول نظام الدليل (DSP): أحكام بروتوكول DSP عندما يكون أداء DSA مختلفاً عن DSA موثق، وبالتالي تتحقق النتائج المتصفحة بواسطة المنفذ الأولى.

8.7.3 خطأ: معلومات أرسلت من المنفذ إلى الطالب تحمل ناتج سلبي لطلب استقبل في السابق.

9.7.3 خطأ جسيم: خطأ محدد يدل على أن العملية لا يمكن أداؤها حالياً دون تدخل خارجي.

10.7.3 إسناد تشغيلي تراتبي (HOB): العلاقة بين وكيلي نظام دليل يحتفظان بسياقات تسمية، واحد تابع لآخر، يحتفظ DSA الرئيسي فيه بمرجع تابع لـــ DSA تابع.

11.7.3 منفذ أولي: يبدأ DSA الأول أداء عملية، أي، يدخل DSA الأول مرحلة تقييم العملية.

12.7.3 عمليات تعديل: هذه هي عمليات تعديل الدليل، أي، Modify Entry، Add Entry، Remove Entry، Modify DN.

13.7.3 سلسلة متعددة: أسلوب تفاعل يعالج فيه DSA طلب أرسله بنفسه لطلبات متعددة سواء موازية أو تتبعية لترسل إلى DSAs آخرين.

14.7.3 عمليات استفهام لمدخل متعدد: هذه هي عمليات بحث الدليل، أي، List and Search.

15.7.3 ستبانة اسم: عملية تحديد موقع مدخل بواسطة مواءمة تتبعية لكل RDN في اسم مستهدف لذروة DIT.

16.7.3 سناد تشغيلي تراتبي غير محدد (NHOB): العلاقة بين وكيلي نظام دليل رئيسيين يحتفظان بسياقات تسمية، واحد تابع مباشر لآخر، ويحتفظ فيه DSA الرئيسي بمرجع تابع غير محدد لـــ DSA تابع.

17.7.3 جزئة مرجع تابع غير محدد (NSSR): تجزئة مراجع معرفة غير محددة إلى طلبات فرعية لـــ DSAs آخرين لتنفيذها: قد تكون الطلبات الفرعية هذه سواء مسلسلة مع DSAs هذه بواسطة DSA الذي يقوم بالتجزئة، أو مواصلة مرجع يحدد DSAs الذين يعادون إلى الطالب لتنفيذها، أو قد ينفذ DSA المجزء بعض الطلبات الفرعية، تاركاً الأخرى للطالب دون استكشاف لتنفيذها.

18.7.3 قدم العملية: مجموعة من القيم تدل على المدى الذي حدث لاستبانة اسم.

19.7.3 المرسِل: DUA الذي بدأ عملية (موزعة) محددة.

20.7.3 صفح: تعاد نتيجة search أو list في شكل صفحة أو أكثر تتألف من عدد محدود من المداخل.

21.7.3 منفّذ: DSA المستقبل لطلب (أي، لأداء عملية).

ملاحظة – المنفّذ هو أيضاً المنفّذ المبدئي باستثناء العمليات التي تتضمن أكثر من DSA واحداً لتقييمها.

22.7.3 إجراء: مواصفة (غير رسمية) لكيفية تقلبل DSA مع مجموعة ما من متغيرات دخل وشجرة معلومات DSA في نتيجة.

ملاحظة – قد تتطابق متغيرات ونتائج دخل مع المعلومات المستقبلة في عملية مطلوبة والمعلومات المرسلة في الإجابة، أو قد تمثل مراحل وسيطة في حساب إجابة من عملية مطلوبة. ويسمي في 2.14 التنوع السابق لمتغيرات ونتائج دخل خارجي.

23.7.3 إسناد تشغيلي تراتبي ذو علاقة (RHOB): إما HOB أو NHOB، يعتمد على السياق.

24.7.3 مرجعي: ناتج يمكن أن يعيده DSA الذي لا يتمكن من أداء عملية بنفسه، ويحدد DSA واحداً أو أكثر قادراً على أداء العملية.

25.7.3 إجابة: نتيجة أو خطأ.

26.7.3 طلب: معلومات تتألف من شفرة عملية ومتغيرات متصاحبة لنقل عملية دليل من طالب إلى منفذ.

27.7.3 تجزئة طلب: تجزئة طلب إلى طلبات فرعية لـــ DSAs آخرين لتنفيذها: قد تكون الطلبات الفرعية هذه سواء مسلسلة مع DSAs هذه بواسطة DSA الذي يقوم بالتجزئة، أو مواصلة مراجع تحدد DSAs قد تعاد إلى الطالب لتنفيذها، أو قد ينفذ DSA المجزء بعض الطلبات الفرعية، تاركاً الأخرى للطالب دون استكشاف لتنفيذها.

28.7.3 طالب: DUA أو DSA المرسل للطلب لأداء (أي، تنفيذ) عملية.

29.7.3 عمليات استفهامية لمدخل وحيد: هذه هي عمليات قراءة دليل، أي، Read and Compare.

30.7.3 خطأ مبرمج: خطأ قد يكون انتقالياً أو قد يدل على مشكلة محددة الموقع، وعلى أي حالة فإن استخدام مراجع معرفة أو نقطة نفاذ يمكّن من الحصول على النتيجة أو الخطأ الجسيم.

31.7.3 DSA تابع: DSA من اثنين يتقسامان HOB أو NHOB، وDSA المحتفظ بسياق تسمية تابع.

32.7.3 طلب فرعي: طلب مولد من تجزئة طلب.

33.7.3 DSA رئيسي: DSA من اثنين يتقسامان HOB أو NHOB، وDSA المحتفظ بسياق تسمية رئيسي.

34.7.3 DSA رئيسي، تابع: وكيلان رئيسيان لنظام دليل يحتفظان بسياقات تسمية، واحد تابع مباشر لآخر؛ والعلاقة بين الاثنين تدار بوضوح عبر HOB (أو NHOB) أو توجد ضمنياً نتيجة احتفاظ DSA بمرجع تابع (أو تابع غير محدد) لـــ DSA تابع.

35.7.3 اسم شيء مستهدف: اسم مدخل سواء توجه إليه العملية عند مرحلة معينة لاستبانة اسم، أو يشترك في تقييم العملية.

36.7.3 سلسلة وحيدة: أسلوب تفاعل اختياري يستخدمه DSA الذي لا يمكنه أداء عملية بنفسه. ويتسلسل DSA بواسطة تنفيذ عملية DSA آخر ثم يرحل الناتج إلى الطالب الأصلي.

4 مختصرات

لأغراض هذه التوصية | المعيار الدولي، تنطبق المختصرات التالية:

ASN.1 ترميز قواعد التركيب المجرد واحد *(Abstract Syntax Notation)*

DISP بروتوكول تظليل معلومات الدليل *(Directory Information Shadowing)*

DMD ميدان إدارة الدليل *(Directory Management)*

DOP بروتوكول إدارة الاسناد التشغيلي للدليل *(Directory Operational Binding Management)*

DSE مدخل محدد لوكيل نظام الدليل *(DSA-Specific)*

HOB اسناد تشغيلي تراتبي *(Hierarchical Operational)*

NHOB اسناد تشغيلي تراتبي غير محدد *(Non-specific Hierarchical Operational)*

NSSR مرجع تابع غير محدد *(Non-specific Subordinate)*

RHOB إسناد تشغيلي تراتبي ذو صلة *(Non-specific Subordinate)*

5 الاصطلاحات

أعدت مواصفة هذا الدليل، مع استثناءات طفيفة، وفقاً للنص الموحد *لقواعد العرض المعتمد في قطاع تقييس الاتصالات في الاتحاد الدولي للاتصالات واللجنة الكهرتقنية الدولية التابعة للمنظمة الدولية للتوحيد القياسي*، نوفمبر 2001.

ويفهم من مصطلح "مواصفة الدليل" (كما في "مواصفة هذا الدليل") أنه يعني ISO/IEC 9594-4 | ITU-T X.518. ويفهم من مصطلح "مواصفات الدليل" جميع توصيات السلسلة X.500 وجميع أجزاء المعيار الدولي ISO/IEC 9594.

تستعمل مواصفة هذا الدليل عبارة *أنظمة الطبعة الأولى* للإشارة إلى الأنظمة التي تتفق مع الطبعة الأولى لمواصفات الدليل، أي، طبعة عام 1988 من سلسلة توصيات CCITT X.500 وISO/IEC 9594: طبعة عام 1990. وتستعمل مواصفة هذا الدليل عبارة أنظمة الطبعة الثانية للإشارة إلى *الأنظمة* التي تتفق مع *الطبعة الثانية* لمواصفات الدليل، أي، طبعة عام 1993 من سلسلة توصيات ITU-T X.500 وISO/IEC 9594: طبعة عام 1995. وتستعمل مواصفة هذا الدليل عبارة *أنظمة الطبعة الثالثة* للإشارة إلى الأنظمة التي تتفق مع الطبعة الثالثة لمواصفات الدليل، أي، طبعة عام 1997 من سلسلة توصيات ITU-T X.500 وISO/IEC 9594: طبعة عام 1998. وتستعمل مواصفة هذا الدليل عبارة *أنظمة الطبعة الرابعة* للإشارة إلى الأنظمة التي تتفق مع الطبعة الرابعة لمواصفات الدليل، أي، طبعة عام 2001 من سلسلة توصيات ITU-T X.500، X.501، X.511، X.518، X.519، X.520، X.521، X.525، X.530 وطبعة عام 2000 للتوصية ITU-T X.509 والأجزاء 1-10 من ISO/IEC 9594: طبعة عام 2001.

وتستعمل مواصفة هذا الدليل عبارة *أنظمة الطبعة الخامسة* للإشارة إلى الأنظمة التي تتفق مع الطبعة الخامسة لمواصفات الدليل، أي، طبعة عام 2005 من توصيات ITU-T X.500، X.501، X.509، X.511، X.518، X.519، X.520، X.521، X.525، X.530 والأجزاء 1-10 من ISO/IEC 9594: طبعة عام 2005.

تعرض مواصفة هذا الدليل ترميز ASN.1 بحروف Helvetica سوداء. وعند الإشارة إلى نمط وقيم ASN.1 في نص عادى، فهي تتميز عن باقي النص العادى بعرضها بحروف Helvetica سوداء. وتكون أسماء الإجراءات، المشار إليها عند تحديد علم دلالات معالجة، متميزة عن النص العادى بعرضها بحروف Times سوداء. ويعرض السماح بتحكم النفاذ بحروف Times سوداء مائلة.

وإذا كانت البنود في قائمة لها أعداد (مقابل استخدام "–" أو حروف)، تعتبر البنود خطوات في إجراء.

القسم 2 – نظرة شاملة

6 نظرة شاملة

تسمح الخدمة المجردة للدليل باستفسار واسترداد وتعديل معلومات الدليل في DIB. وتوصف هذه الخدمة على أساس شيء مجرد للدليل كما ورد في ISO/IEC 9594-3 | ITU-T X.511. وبالمثل، يسمح بروتوكول LDAP باستفسار واسترداد وتعديل معلومات الدليل في DIB. ويرد هذا البروتوكول والخدمات التي تقر صلاحيته في RFC 3377.

وبالضرورة، فإن مواصفة شيء مجرد للدليل لا تتناول بأي طريقة التحقيق المادي للدليل: ولا تتناول بصورة خاصة مواصفة وكلاء نظام الدليل (DSA) حيث تخزن فيه وتدار، ومن خلاله توفر الخدمة. وفضلاً عن ذلك، لا تعتبر ما إذا كانت DIB مركزية، أي تحتوي في DSA وحيد، أو موزعة عبر عدد من DSAs . وبناءً على ذلك، لا يغطي وصف الخدمة متطلبات DSAs لتتوفر لهم معرفة وللملاحة والتعاون مع DSAs آخرين، لدعم الخدمة المجردة في بيئة موزعة.

تحدد مواصفة الدليل هذا تنقيح شيئ محدد للدليل، ويجرى التعبير عن التنقيح على أساس مجموعة من أشياء DSA واحداً أو أكثر تشكل مجتمعة خدمة دليل موزعة.

وبالإضافة إلى ذلك، تحدد مواصفة الدليل هذا الطرائق المسموح بها التي يمكن فيها توزيع DIB عبر DSA واحداً أو أكثر. وفي الحالة المحدودة حيث يجرى احتواء DIB في DSA وحيداً، يكون الدليل في الحقيقة مركزياً؛ وفي الحالة حيث DIB موزعة عبر DSA واحداً أو أكثر، تحدد آليات المعرفة والملاحة لتضمن أن كامل DIB يمكن النفاذ إليها من جميع DSAs الذين يحتفظون بمداخل مكونة.

ويمكن استنساخ أجزاء من DIB في DSAs متعددين. وتسمح البروتوكولات الواردة في مواصفة الدليل هذا باستخدام معلومات مستنسخة لتحسين توافر وأداء وكفاءة خدمة الدليل الموزعة. ويكون استخدام المعلومات المستنسخة، إلى حد ما، تحت تحكم المستعمل، من خلال استخدام خيارات مراقبة الخدمة. وتدل أيضاً الإجراءات الواردة في مواصفة الدليل هذا على بعض فرص تصميم مستمثلات عند استخدام معلومات مستنسخة.

وبالإضافة إلى ذلك، ترد تفاعلات مناولة طلب تمكن من إقرار صلاحية خواص تشغيلية معينة للدليل ليتحكم فيها مستعمليه. وبشكل خاص، يكون للمستعمل تحكم على ما إذا كان DSA، المستجيب لاستفهام دليل يتعلق بمعلومات يحتفظ بها DSA آخر، له خيار استجواب DSA (DSAs) مباشر (سلسلة) آخر يمكنه أن يتقدم بالاستفهام (مرجع).

وعامة، فإن قرار DSA بالتسلسل أو الاحالة يعتمد على مراقبة الخدمة التي يحددها المستعمل، وبواسطة الظروف الإدارية أو التشغيلية أو التقنية لـــ DSA.

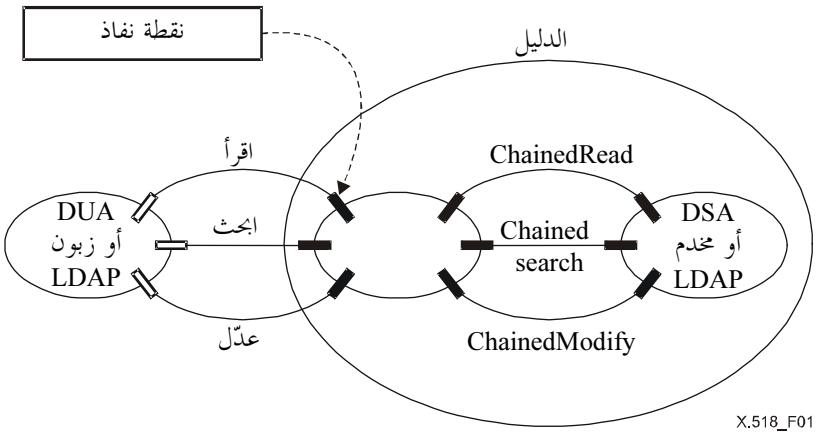
وعند إدراك ذلك، عامة، يجرى توزيع الدليل، وتلبي استفسارات ذلك الدليل بواسطة عدد اعتباطي لـــ DSAs متعاونين يقومون إعتباطياً بالتسلسل أو الإحالة طبقاً للمعايير الواردة أعلاه، وتورد مواصفة الدليل هذا الإجراءات الملائمة التي ينفذها DSAs استجابة لاستفسارات دليل موزع. وتضمن هذه الإجراءات أن مستعملي خدمة الدليل الموزع يدركون أنها صديقة المستعمل ومتماسكة.

القسم 3 – نماذج دليل موزع

7 نموذج نظام دليل موزع

تضع الخدمة المجردة للدليل، كما تعرف في ITU-T X.511 | ISO/IEC 9594-3، نموذج الدليل باعتباره شيئاً يوفر مجموعة من خدمات الدليل لمستعمليه. وينفذ مستعملو الدليل إلى خدماته من خلال نقطة نفاذ. وقد يكون للدليل نقطة نفاذ واحدة أو أكثر وتتميز كل نقطة نفاذ بالخدمات التي تقدمها وأسلوب التفاعل المستخدم لتوفير هذه الخدمات.

يوضح الشكل 1 نموذج الدليل الموزع الذي يستخدم كأساس لتحديد الجوانب الموزعة للدليل. ويوضح الدليل على أنه مجموعة تتألف من DSA واحداً أو أكثر.



الشكل 1 – أشياء نموذج دليل موزع

يحدد DSAs بالتفصيل في الأقسام التالية من مواصفة الدليل هذا. ويقر هذا القسم عدداً من خواصهم للقيام بدور مقدمة واقامة علاقة بين مواصفة الدليل هذا ومواصفات الدليل الأخرى.

ويعرف DSAs من أجل استيعاب توزيع DIB ولكي يتفاعل عدد من DSAs الموزعين مادياً بطريقة فرضية وتعاونية لتوفير خدمات الدليل لمستعملي الدليل (DUAs أو زبائن LDAP).

يوضح الشكل 1 العلاقة بين الخدمة المجردة للدليل والخدمة المجردة لـــ DSA. وتوفر الخدمة المجردة للدليل المعرفة في ITU-T X.511 | ISO/IEC 9594-3 من خلال عدد من عمليات الدليل. ولتحقيق هذه الخدمة، يتفاعل DSAs الذين يؤلفون الدليل مع كل منهما. ويعرف طابع هذا التفاعل على أساس الخدمة التي قد يقدمها DSA واحداً إلى DSA آخر، الخدمة المجردة لـــ DSA. وتوفر الخدمة المجردة لـــ DSA من خلال عدد من العمليات، تسمي عمليات مسلسلة، يكون لكل منها نظير في الخدمة المجردة للدليل. ومن ثم، فإن عملية ما في خدمة مجردة للدليل، مثلاً Read، قد تتطلب أن يتفاعل DSA الموفر للخدمة مع DSA واحداً أو أكثر باستخدام عمليات متسلسلة، مثلاً Chained Read.

ملاحظة – قد يكون من الممكن لـــ DSAs الطالبين لـــ LDAP تسلسل عمليات، مثلاً، باستخدام LDAP Controls أو Extended Operations؛ ومع ذلك، فإن الإجراءات والبروتوكولات الضرورية لتحقيق هذا هي خارج مدى مواصفة الدليل هذا.

8 نموذج تفاعلات DSA

إن الخاصية الأساسية للدليل هي أن المستعمل، في حالة DIB موزعة، ينبغي أن يكون قادراً على الحصول على أي طلب خدمة بشكل مرضي (على أن يخضع للأمن ومراقبة النفاذ والسياسات الإدارية) بغض النظر عن نقطة النفاذ التي يبدأ منها الطلب. ولاستيعاب هذا المتطلب، من الضروري لأي DSA عامل في تلبية طلب خدمة معينة أن تتوفر له معرفة (كما ورد في

ISO/IEC 9594-2 | ITU-T X.501) حيث يحدد موقع المعلومات المطلوبة وسواء تعاد هذه المعلومات إلى الطالب أو محاولة تلبية الطلب نيابة عنه. (قد يكون الطالب DUA أو زبون LDAP أو DSA آخر: وفي الحالة الأخيرة يدعم كل من DSAs DSP).

تعرف ثلاثة نماذج لتفاعل DSA لتلبية هذه المتطلبات، أي "سلسلة وحيدة" و"سلسلة متعددة" و"مرجع". وفي باقي مواصفة الدليل هذا، يستخدم المصطلح التنوعي سلسلة ليشير إلى سلسلة وحيدة و/أو سلسلة متعددة حسب الاقتضاء في السياق. وتشير "سلسلة" إلى محاولة DSA لتلبية طلب أرسلته عملية متسلسلة واحدة أو أكثر إلى DSAs آخرين؛ ويشير "مرجع" إلى إعادة معلومات معرفة إلى الطالب، الذي قد يتفاعل نفسه مع DSA أو أكثر معرف في معلومات المعرفة.

قد ينتج تفاعل سلسلة وحيدة أو مرجع من طلب وحيد. وبديلاً عن ذلك، قد يجزأ الطلب إلى طلبات فرعية عديدة قبل التفاعل. وقد ينتج عن تفاعل سلسلة متعددة أو مرجع، أو خليطاً من الاثنين، من طلب مجزء. ويعرف نمطان من التجزئة؛ تجزئة NSSR وتجزئة طلب.

1.8 تجزئة طلب

1.1.8 تجزئة NSSR

إن تجزئة NSSR هي عملية إعداد طلبات مماثلة جاهزة للنقل (سواء متتابعة أو بالتوازي) إلى DSAs تابعين عديدين نتيجة لمواجهة NSSR خلال استبانة اسم. ولا تحتفظ مراجع تابعة غير محددة بـــ RDNs لسياقات تسمية تابعة مرجعية، بحيث إن DSA المرجعي غير قادر على التعرف على أي DSA تابع يحتفظ بأي سياق (سياقات) تسمية تابع. وخلال استبانة اسم، يرسل DSA المواجه لـــ NSSRs طلب ماثل إلى كل DSA تابع (في غياب تظليل). ويمكن أن يتم هذا بالتتابع أو بالتوازي. ونمطياً، يتمكن DSA واحد فقط من مواصلة استبانة اسم؛ ويعيد الآخرون **serviceError** مع مشكلة **unableToProceed**. وفي ظروف معينة (نادرة) من الممكن أن يواصل أكثر من DSA استبانة اسم، موضحا نتائج مستنسخة.

ملاحظة – لا يمكن لـــ NSSRs الاشارة إلى خدومي LDAP.

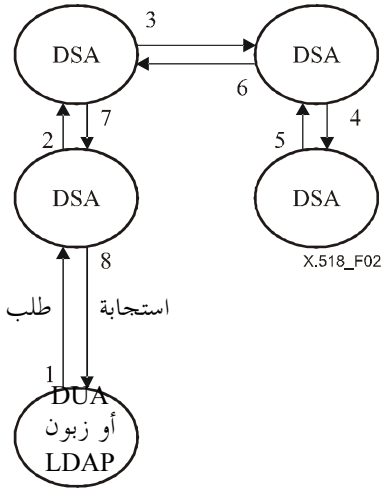
2.1.8 تجزئة طلب

إن تجزئة طلب، وهو الشكل الآخر لتجزئة طلب، هي عملية يؤدى بها DSA داخلياً قبل الاتصال بـــ DSA واحد أو أكثر آخر و/أو خدومي LDAP. ويجزء طلب إلى عديد من الطلبات الفرعية، من الممكن مختلفة، بحيث إن كل طلب فرعي ينجز جزء من المهمة الأصلية. ويمكن استخدام تجزئة طلب فقط خلال تقييم عملية List أو Search. وبعد تجزئة طلب، يمكن تسلسل الطلبات الفرعية مع DSAs آخرين و/أو خدومي LDAP لمواصلة المهمة، أو يمكن إعادة نتيجة جزئية (مرجع مدمج) إلى الطالب. ومثال لنفس طلب فرعي يتولد لـــ DSAs آخرين و/أو خدومي LDAP هو عندما يكون لمدخل مراجع تابعة و/أو NSSRs تشير معاً إلى DSA واحد أو خدوم LDAP واحد. ومثال للطلبات الفرعية المختلفة التي تولد لنفس DSAs و/او خدومي LDAP مختلفين عندما يواجه مدخلان مختلفان خلال Search (شجرة فرعية) ولكل منها مرجع تابع.

2.8 سلسلة وحيدة

يمكن استخدام أسلوب التفاعل هذا (موضح في الشكل 2) بواسطة DSA واحد لتمرير طلب إلى DSA آخر عندما يكون للأول معرفة عن سياقات تسمية يحتفظ بها الأخير. ويمكن أن تستخدم السلسلة الوحيدة للاتصال بـــ DSA وحيد مشيراً إلى إسناد مرجعي أو مرجع تابع أو مرجع علوي أو مرجع مورد أو مرجع رئيسي.

ملاحظة – في الشكل 2، يعرف ترتيب التفاعلات بواسطة الأعداد المتصاحبة مع خطوط التفاعل.



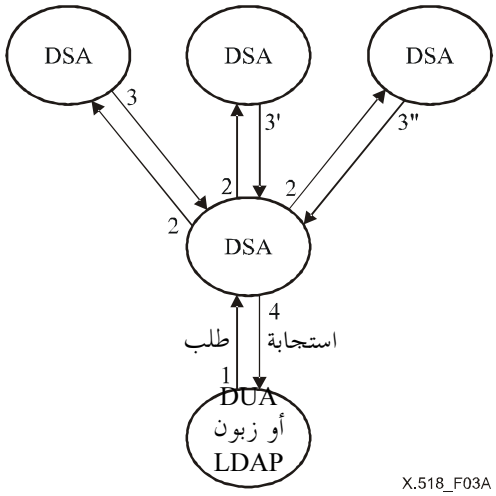
الشكل 2 – أسلوب سلسلة وحيدة

3.8 سلسلة متعددة

يستخدم DSA أسلوب التفاعل هذا لنقل طلبات عديدة جارية نتجت عن طلب قادم، نتيجة إما لتجزئة طلب أو تجزئة NSSR.

1.3.8 سلسلة متعددة موازية

مع سلسلة متعددة موازية، ينقل DSA طلبات عديدة جارية في نفس الوقت (انظر الشكل 3 أ). وبينما السلسلة المتعددة الموازية قد تحسن الأداء، قد تسبب تحت ظروف معينة، مثل، وجود تظليل، استنساخ نتائج تستقبل.

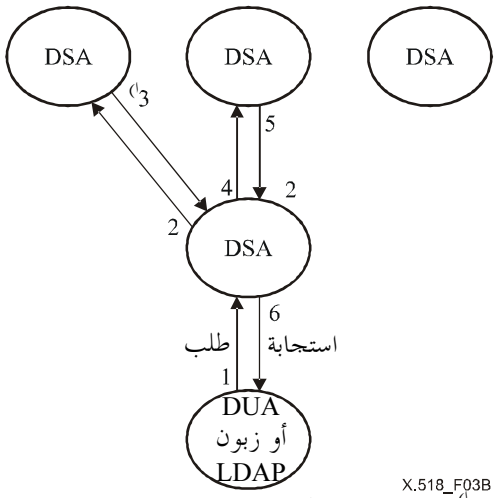


الشكل 3 أ – سلسلة متعددة موازية

2.3.8 سلسلة متعددة تتابعية

مع سلسلة متعددة تتابعية، ينقل DSA طلب جاري واحد في وقت واحد وينتظر نتيجة أو خطأ لطلب واحد قبل إرسال التالي (انظر الشكل 3 ب). وبينما السلسلة المتعددة التتابعية قد لا تكون الأسلوب الأسرع للتفاعل، من غير المحتمل استنساخ نتائج تستقبل.

ملاحظة – قد يستخدم DSA تركيباً من سلسلة متعددة موازية وسلسلة متعددة تتابعية.



[1] غير قادر على المواصلة

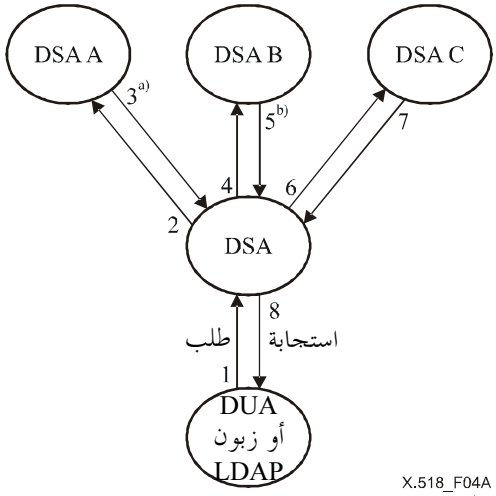
الشكل 3 ب – سلسلة متعددة تتابعية
(نتيجة لتجزئة NSSR)

4.8 مرجع

يعيد DSA مرجع (موضح في الشكلين 4 أ و4 ب) استجابة لطلب من DUA أو زبون LDAP أو DSA آخر. وقد يشكل المرجع استجابة كاملة (وفي هذه الحالة يعتبر خطأ) أو مجرد جزء من الاستجابة. ويحتوى المرجع على مرجع معرفة، الذي يكون إما أعلى أو تابع أو تابع غير محدد أو مورد أو مرجع رئيسي.

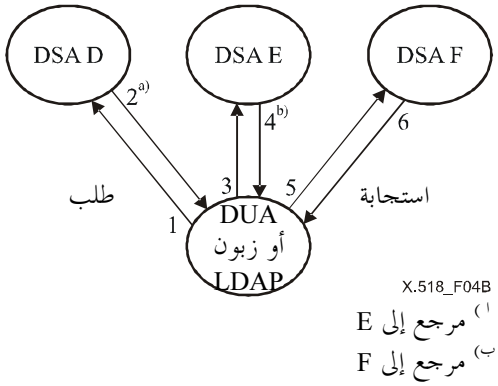
قد يستخدم DSA (الشكل 4 أ) المستقبل للمرجع مرجع المعرفة المحتوى عليه، لسلسلة أو سلسلة متعددة فيما بعد (يعتمد على نمط المرجع) الطلب الأصلي لـــ DSAs آخرين. وبديلاً عن ذلك، يمكن لـــ DSA المستقبل لمرجع أن يمرر بدوره المرجع في استجابته. ويمكن لـــ DUA أو زبون LDAP (الشكل 4 ب) المستقبل لمرجع أن يستخدمه للاتصال بـــ DSA واحد أو أكثر للتقدم بالطلب.

ملاحظة – في الشكلين 4 أ و4 ب، يعرف ترتيب التفاعلات بواسطة الأعداد المتصاحبة مع خطوط التفاعل.



[أ] مرجع إلى B
[ب] مرجع إلى C

الشكل 4 أ – أسلوب مرجع (يعمل DSA على مراجع)



الشكل 4 ب – أسلوب مرجع (يعمل DUA على مراجع)

5.8 تحديد الأسلوب

إذا لم يستطع DSA حل الطلب بالكامل بنفسه، يسلسل الطلب (أو طلب مشكل بواسطة تجزئة الطلب الأصلي) بـــ DSA آخر، ما لم:

أ) تحظر السلسلة من قبل المستعمل عبر مراقبة خدمة، وفي تلك الحالة يعيد DSA مرجع أو **serviceError** مع مشكلة **chainingRequired**؛ أو

ب) يكون لـــ DSA أسباب إدارية أو تشغيلية أو تقنية لعدم الإحالة إلى السلسلة، وفي تلك الحالة يعيد DSA مرجع.

الملاحظة 1 – "السبب التقني" لعدم التسلسل هو أن DSA المعرف في مرجع المعرفة لا يدعم DSP.

الملاحظة 2 – إذا ضبطت مراقبة خدمة **localScope**، فإن DSA (أو DMD) إما أن يحل الطلب أو يعيد خطأ.

الملاحظة 3 – إذا فضل المستعمل مراجع، ينبغي على المستعمل أن يضبط **chainingProhibited**.

القسم 4 – الخدمة المجردة لـــ DSA

9 نظرة شاملة على الخدمة المجردة لـــ DSA

يرد وصف كامل لخدمة الدليل في ISO/IEC 9594-3 | ITU-T X.511. وعند توفير هذه الخدمة في بيئة موزعة، كما ورد نموذجها في القسم 7، يمكن اعتبارها أنها تتوفر بواسطة مجموعة من DSAs. ويوضح ذلك الشكل 1.

لكل عملية معرفة في خدمة الدليل، تُعرف عملية "متسلسلة" متوافقة في الخدمة المجردة لـــ DSA لاستخدامها بين DSAs متعاونين لإنجاز عملية خدمة الدليل تلك. ومن ثم، قد يطلب DSA يستقبل عملية Read من DUA مساعدة DSA آخر (مثل DSA يحتفظ بمدخل مستهدف أو نسخة منه) لتلبيتها، ولذا يرسل إلى DSA ذلك عملية Chained Read.

تعرف أنماط المعلومات المتبادلة في الخدمة المجردة لـــ DSA في القسم 10. وتعرف عمليات وأخطاء الخدمة المجردة لـــ DSA في الأقسام من 11 إلى 13.

10 أنماط المعلومات

1.10 مقدمة

يحدد هذا القسم، وفي بعض الحالات يعرف، عدداً من أنماط المعلومات التي تستخدم فيما بعد في تعريف عمليات مختلفة لخدمة مجردة للدليل. وأنماط المعلومات المعنية هي الشائعة في أكثر من عملية واحدة، من المحتمل أن تكون في المستقبل أو المعقدة بما فيه الكفاية أو المكتفية ذاتياً بحيث تستحق تعريفها بشكل منفصل عن العمليات التي تستخدمها.

إن العديد من أنماط المعلومات المستخدمة في تعريف خدمة مجردة للدليل هي المعرفة فعلاً في أماكن أخرى. ويحدد القسم الفرعي 2.10 هذه الأنماط ويدل على مصدر تعريفها. وتحدد الأقسام الفرعية من 3.10 إلى 10.10 وتعرف نمط معلومات.

2.10 أنماط المعلومات المعرفة في أماكن أخرى

تعرف أنماط المعلومات التالية في ISO/IEC 9594-2 | ITU-T X.501:

- **aliasedEntryName**؛
- **DistinguishedName**؛
- **Name**؛
- **RelativeDistinguishedName**.

تعرف أنماط المعلومات التالية في ISO/IEC 9594-3 | ITU-T X.511:

(ربط)

- **DirectoryBind**

(عمليات)

- **Abandon**

(أخطاء)

- **abandoned**؛
- **attributeError**

– **nameError**؛

– **securityError**؛

– **serviceError**؛

– **updateError**.

(صنف شيء معلومات)

– **OPTIONALLY-PROTECTED**

(نمط معطيات)

– **SecurityParameters**

يعرف نمط المعلومات التالي في ITU-T X.520 | ISO/IEC 9594-6:

– **PresentationAddress**.

3.10 متغيرات سلسلة

تحين **ChainingArguments** في كل عملية متسلسلة، لتنقل إلى DSA المعلومات المطلوبة للأداء الناجح لجزئه من المهمة الشاملة:

```
ChainingArguments ::= SET {
        originator                [0]   DistinguishedName OPTIONAL,
        targetObject              [1]   DistinguishedName OPTIONAL,
        operationProgress         [2]   OperationProgress
                                        DEFAULT { nameResolutionPhase notStarted },
        traceInformation          [3]   TraceInformation,
        aliasDereferenced         [4]   BOOLEAN DEFAULT FALSE,
        aliasedRDNs               [5]   INTEGER OPTIONAL,
                        -- only present in first edition systems
        returnCrossRefs           [6]   BOOLEAN DEFAULT FALSE,
        referenceType             [7]   ReferenceType DEFAULT superior,
        info                      [8]   DomainInfo OPTIONAL,
        timeLimit                 [9]   Time OPTIONAL,
        securityParameters        [10]  SecurityParameters DEFAULT { },
        entryOnly                 [11]  BOOLEAN DEFAULT FALSE,
        uniqueIdentifier          [12]  UniqueIdentifier OPTIONAL,
        authenticationLevel       [13]  AuthenticationLevel OPTIONAL,
        exclusions                [14]  Exclusions OPTIONAL,
        excludeShadows            [15]  BOOLEAN DEFAULT FALSE,
        nameResolveOnMaster       [16]  BOOLEAN DEFAULT FALSE,
        operationIdentifier       [17]  INTEGER OPTIONAL,
        searchRuleId              [18]  SearchRuleId OPTIONAL,
        chainedRelaxation         [19]  MRMapping OPTIONAL,
        relatedEntry              [20]  INTEGER OPTIONAL,
        dspPaging                 [21]  BOOLEAN DEFAULT FALSE,
        nonDapPdu                 [22]  ENUMERATED { ldap (0) } OPTIONAL,
        streamedResults           [23]  INTEGER OPTIONAL
        excludeWriteableCopies    [24]  BOOLEAN DEFAULT FALSE }

Time ::= CHOICE {
        utcTime           UTCTime,
        generalizedTime   GeneralizedTime }
```

يكون للمكونات المختلفة المعاني المعرفة أدناه:

أ) ينقل مكون **originator** اسم المرسل (النهائي) للطلب ما لم يكن محدداً فعلاً في معلمات الأمن. وإذا كان **requester** محيناً في **CommonArguments**، قد يحذف هذا المتغير.

الملاحظة 1 – إذا كان للمرسل أسماء بديلة تفاضلية حسب السياق، فإن الاسم المستخدم باعتباره قيمة الـــ **originator** يكون الاسم المميز الأولي، إذا عرف. وإلا، لا يعمل الاستيقان ونقطة النفاذ القائمين على قيمة **originator** كما يراد.

ب) ينقل مكون **targetObject** اسم شيء يجرى تسيير مدخل دليله. ويعتمد دور هذا الشيء على العملية الخاصة المعنية: قد يكون شيء يجرى تشغيل مدخله، أو يكون قاعدة لطلب أو طلب فرعي يتضمن أشياء متعددة (مثل، **chainedList** أو **chainedSearch**). ويمكن حذف هذا المكون فقط إذا كان له نفس قيمة الشيء أو معلمة قاعدة شيء في عملية متسلسلة، وفي تلك الحالة تكون قيمته المتضمنة هي تلك القيمة.

وعندما يشمل **targetObject** RDNs تحتوى على نمط نعت وأزواج قيمة يكون لها قيم مميزة متعددة تفاضلية حسب السياق، تكون RDNs التي تم استبانتها هي RDNs الأولية.

ج) يستخدم مكون **operationProgress** لإبلاغ DSA بالتقدم المحرز في العملية، ومن ثم الدور المتوقع أن يقوم به في أدائه الشامل. وترد المعلومات المنقولة في هذا المكون في 5.10.

د) يستخدم مكون **traceInformation** لمنع العروات فيما بين DSAs عندما يكون تسلسل في العملية. ويضيف DSA عنصراً جديداً إلى معلومات أثر قبل تسلسل عملية بـــ DSA آخر. وعند طلب أداء عملية، يتأكد DSA، بواسطة فحص معلومات الأثر، بأن العملية لم تشكل عروة. وترد المعلومات المنقولة في هذا المكون في 6.10.

هـ) يكون مكون **aliasDereferenced** هو قيمة **BOOLEAN** التي تستخدم للدلالة عما إذا كان مدخلاً واحداً أو أكثر أم لا لم تتم مواجهته وتبديله خلال استبانة اسم موزع. وتدل القيمة بالتغيب لـــ **FALSE** على عدم إبدال مدخل مستعار.

و) يدل مكون **aliasedRDNs** على عدد RDNs في **targetObject Name** تم توليدها من نعوت **aliasedEntryName** لمدخل مستعار واحد (أو أكثر). وتضبط القيمة الصحيحة عند مواجهة وإبدال مدخل مستعار. ويكون هذا المكون محيناً إذاً، وإذا فقط كان مكون **aliasDereferenced** هو **TRUE**.

الملاحظة 2 – يتوفر هذا المكون من أجل تلاؤم تنفيذ الطبعة الأولى من الدليل. ويقوما DUAs (و DSAs) المنفذين طبقاً لطبعات لاحقة من مواصفات الدليل دائماً بحذف المعلمة من **CommonArguments** للطلب التالي. وبهذه الطريقة، لا يشور الدليل خطأ إذا قام المستعارون بإبدال مستعارين آخرين.

ز) إن مكون **returnCrossRefs** هو قيمة Boolean تدل ما إذا كانت أم لا مراجع المعرفة، المستخدمة خلال أداء عملية موزعة، مطلوبة لتمريرها إلى الخلف إلى DSA ابتدائي باعتباره إسناد مرجعي، مع نتيجة أو مرجع. وتدل القيمة بالتغيب لـــ **FALSE** على أن مراجع المعرفة تلك لا تعاد.

ح) يدل مكون **referenceType**، بالنسبة لـــ DSA يطلب منه أداء العملية، أي نمط من المعرفة استخدم لتسيير الطلب إليه. ولهذا يمكن لـــ DSA أن يكتشف الأخطاء في المعرفة التي يحتفظ بها المنفذ. وإذا اكتشف مثل هذا الخطأ، يدل عليه **serviceError** مع مشكلة **invalidReference**. ويرد وصف كامل لـــ **ReferenceType** في 7.10.

الملاحظة 3 – إذا لم يوجد **referenceType**، تفترض القيمة **superior**.

ط) يستخدم مكون **info** لنقل معلومات محددة – DMD فيما بين DSAs التي تعمل في معالجة طلب مشترك. وهذا المكون من النمط **DomainInfo**، وهو من النمط المسموح به:

```
DomainInfo ::= ABSTRACT-SYNTAX.&Type
```

ي) يدل المكون **timeLimit**، إذا حيّن، على الوقت التي تنتهي فيه العملية (انظر 1.4.1.16). وقبل أن تستخدم قيمة **Time** في أي عملية مقارنة وإذا كانت قواعد تركيب **Time** قد اختيرت باعتبارها نمط **UTCTime**، تسوى قيمة مجال السنة ذات الرقمين إلى قيمة سنة ذات أربعة أرقام كما يلي:

- إذا كانت قيمة الرقمين 00 حتى 49 شاملة، تضاف إلى القيمة 2000 إليها.
- إذا كانت قيمة الرقمين 50 حتى 99 شاملة، تضاف إلى القيمة 1900 إليها.

الملاحظة 4 – قد يمنع استخدام **GeneralizedTime** التشغيل البيني مع تنفيذ دون وعي لإمكانية اختيار إما **UTCTime** أو **GeneralizedTime**. إنها مسؤولية من يحددون الميادين التي تستخدم فيها مواصفة هذا الدليل، مثل، زمرات مظهر جانبي، عندما يستخدم **GeneralizedTime**. ولا يمكن استخدام **UTCTime** في أي حال من الأحوال لتمثل تواريخ بعد 2049.

ك) يرد مكون **SecurityParameters** في ISO/IEC 9594-3 | ITU-T X.511. ويعتبر غيابة مماثل لمجموعة فارغة من معلمات أمن.

ل) يضبط مكون **entryOnly** على **TRUE** إذا كانت العملية الأصلية هي Search مع متغير **subset** مضبوط على **oneLevel**، وتمت مواجهة مدخل مستعار كتابع مباشر لـــ **baseObject**. ويؤدى DSA الذي يؤدى بنجاح استبانة اسم على اسم **targetObject** تقييم شيء على مدخل مسمي فقط.

م) يورد اختيارياً مكون **uniqueIdentifier** عندما يطلب لتأكيد اسم المرسل. ويرد وصف نمط معطيات **uniqueIdentifier** في ITU-T X501/ISO/IEC 9594-2.

ن) يورد اختيارياً مكون **authenticationLevel** عندما يطلب ليدل على الطريقة التي نفذ بها الاستيقان. ويرد وصف نمط معطيات **authenticationLevel** في ITU-T X501/ISO/IEC 9594-2.

س) يكون لمكون **exclusions** أهمية فقط لعمليات Search؛ ويدل، إذا حين، على أي أشجار فرعية لمداخل تابعة لـــ **targetObject** تستبعد من نتيجة عملية Search (انظر 9.10).

ع) يكون لمكون **excludeShadows** أهمية لعمليات Search وList فقط؛ ويدل على أن البحث ينطبق على مداخل وليس على نسخ مداخل. ويمكن استخدام المكون الاختياري هذا من قبل DSA كطريقة لتجنب استقبال نتائج مستنسخة (انظر 1.20).

ف) يكون لمكون **nameResolveOnMaster** أهمية فقط خلال استبيان اسم، ويضبط فقط إذا تمت مواجهة NSSRs. وإذا ضبط على **TRUE**، يشير إلى استبانة الاسم التالي، أي، مواءمة RDNs المتبقية من **nextRDNToBeResolved**، ولا يستخدم معلومات نسخة مدخل، بما في ذلك نسخاً يمكن كتابتها في تنفيذ رئيس متعدد؛ وتتم الاستبانة التالية لكل RDN متبقي في DSA الرئيسي للمدخل الذي حدده RDN ذلك (انظر 1.20).

ص) ييسر مكون **operationIdentifier** ترابط عمليات DAP مع عمليات DSP التالية ذات العلاقة وكذلك مع النتائج. ويعينه DSA الذي يستقبل أولاً طلب DAP أو ينسخ من متغيرات سلسلة لطلبات DSP التي تتطلب مزيداً من التسلسل. ولا يعيد DSA الذي يعين **operationIdentifier** استخدام الرقم الصحيح المعين لفترة طويلة من الزمن كافية. وييسر ترابط طلبات ونتائج DAP وDSP بواسطة تسجيل DSA، لكل عملية ونتيجة، و**operationIdentifier** مع اسم DSA المعين له (أول DSA في **traceInformation** على طلب متسلسل). وقد يكون ذلك الترابط مفيداً لأغراض التسجيل والتدقيق والتسلسل والتسويات وما إلى ذلك.

ق) ينقل مكون **searchRuleId** الهوية الفريدة لـــ search-rule. وهو متضمن بواسطة DSA المؤدى لـــ **Search procedure (I)** الابتدائي في حالة بدء هذا الإجراء في خدمة منطقة إدارية محددة وتتقدم عملية

البحث في DSAs آخرين سواء عند التقدم الهابط DIT، وعند اتباع مستعارين أو عند اتباع مؤشرات زمرة تراتبية.

ر) يقر مكون **chainedRelaxation** صلاحية تنفيذ إرخاء بطريقة موزعة لعمليات بحث متسلسلة. وإذا استقبل DSA عملية بحث متسلسلة، ودعم سياسات إرخاء، يمكن أن يستخدم مكون **chainedRelaxation** المورد بدلا من أي سياسة إرخاء أخرى قد تنفذ، وبالتالي يمكن من تنسيق الإرخاء فيما بين DSAs الذين يحتمل أن يعيدوا نتائج بحث.

ش) يكون عنصر **relatedEntry** محيناً عندما يطلب DSA المستقبل لاستبانة مداخل ذات علاقة. وعند التحيين، يستجيب DSA المستقبل فقط لعنصر مدخل محدد له علاقة تحدده قيمة **relatedEntry** في **joinAttributes** لـــ **SearchArgument**. ومن ثم، تختار قيمة صفر لـــ **relatedEntry** أول عنصر في تتابع **joinAttributes** لـــ **SearchArgument**. ولا تتجاوز القيمة واحد أقل من عدد العناصر في مكون joinAttributes لـــ **SearchArgument**. وغياب عنصر relatedEntry في **ChainingArguments** لعملية DSP المحدد لمداخل ذات علاقة يشير إلى أن العملية الموزعة التي يجرى تسلسلها هي بحث قاعدة وليس جزءاً من المدخل المتعلق بالبحث.

الملاحظة 5 – إذا طلب DSA الذي ينفذ له تسلسل لمناولة كل من نتائج بحث عادي ونتائج مدخل متعلق، يتم هذا بواسطة إرسال DSA لعمليتين لـــ DSP مميزتين.

وعندما يكون عنصر **relatedEntry** محيناً، تنطبق القواعد الخاصة التالية:

- عند تقييم المكون الفرعي لـــ **infoTypes** لـــ مكون **selection** لـــ **SearchArgument**، يؤخذ **infoTypes** على أن له قيمة **attributeTypesAndValues**، كيفما كانت القيمة محددة في الأصل؛
- إن جميع النعوت المحددة في أي مكون **joinAtt** لـــ **JoinAttPair** تكون متضمنة في الاختيار، بغض النظر عما إذا كانت متضمنة في السابق أم لا؛
- يحذف DSA المنسق لنتائج مدخل ذات علاقة القيم والمتغيرات غير المحددة، حتى يجعل النتيجة تتطابق مع طلب المستعمل الأصلي.

ويمرر متغير relatedEntry في تتابع خارج **ChainingArguments** بواسطة DSA الذي يدعم مداخل ذات علاقة.

ت) إذا كان DSA الموثق مختلفاً عن المنفذ الابتدائي (انظر 5.5.15) ويدعم DSA الموثق النتائج المتصفحة لـــ DSP، قد يضبت هذا المكون على **TRUE** ليوجه المنفذ الابتدائي لتوفير نتائج متصفحة لـــ DSP. وإذا كان هذا المكون هو **FALSE** (بالتغيب)، لا يؤدى المنفذ الابتدائي نتائج متصفحة لـــ DSP. إن المنفذ الابتدائي الذي يدعم نتائج متصفحة لـــ DSP لا يرسل هذا المكون إلى الأمام إلى (s) DSA الذي يرسل طلبات فرعية.

ث) يستخدم مكون **nonDapPdu** للدلالة إذا كان PDU مغلف في متغير متسلسل ناشيء من طلب غير Dap مثل طلب LDAP.

خ) يستخدم مكون **streamedResults** كعداد لتحديد ما إذا كان من الممكن تسلسل النتائج المتدفقة في استجابة لهذه العملية. ويزيد كل DSA متضمن في Name Resolution العداد بواحد إذاً، وإذا فقط كان العداد محين، ويفهم DSA النتائج المتدفقة، وDSA على استعداد لقبول نتائج متدفقة للعملية المتسلسلة هذه. ثم هذا العداد يستخدم DSA الذي ينهي Name Resolution لتحديد ما إذا كان كل DSA سابق معد لمناولة نتائج متدفقة.

ذ) يكون لمكون **excludeWriteableCopies** أهمية فقط لعمليات Search وList؛ ويدل على أن البحث ينطبق على النسخ الرئيسية الأولية للمداخل وليس على النسخ الممكن كتابتها لتلك المداخل. ويمكن أن يستخدم DSA المكون الاختياري هذا كطريقة لتجنب استقبال نتائج مستنسخة (انظر 1.20).

4.10 نتائج سلسلة

تكون **ChainingResults** محينة في نتيجة كل عملية وتوفر تغذية راجعة لـــ DSA الذي ينفذ العملية.

```
ChainingResults ::= SET {
        info                [0]  DomainInfo OPTIONAL,
        crossReferences     [1]  SEQUENCE SIZE (1..MAX) OF CrossReference OPTIONAL,
        securityParameters  [2]  SecurityParameters DEFAULT { },
        alreadySearched     [3]  Exclusions OPTIONAL }
```

يكون للمكونات المختلفة المعاني المعرفة أدناه:

أ) يستخدم مكون **info** لنقل معلومات محددة – DMD فيما بين DSAs التي تعمل في معالجة طلب مشترك. وهذا المكون من نمط **DomainInfo**، وهو من النمط المسموح به.

ب) لا يحين مكون **crossReferences** في **ChainingResults** ما لم يكن لمكون **returnCrossRefs** للطلب المتطابق قيمة **TRUE**. ويتألف هذا المكون من تتابع بنود **CrossReference**، يحتوى كل منها على **contextPrefix** وواصف **accessPoint** (انظر 8.10).

```
CrossReference ::= SET {
    contextPrefix     [0]  DistinguishedName,
    accessPoint       [1]  AccessPointInformation }
```

قد يضاف **CrossReference** من قبل DSA عندما يوائم جزء من متغير **targetObject** لعملية مع سابقة واحدة من سابقات سياق. وقد يكون للسلطة الإدارية لـــ DSA سياسة لا تعيد مثل تلك المعرفة، وفي هذه الحالة، لا تضيف بندا إلى التتابع.

ج) يرد مكون **SecurityParameters** في ISO/IEC 9594-3 | ITU-T X.511. ويعتبر غياب مكون **securityParameters** مماثل لمجموعة فارغة من معلمات أمن.

د) يدل مكون **alreadySearched**، إذا حين، أي تابع لـــ RDNs تابع لـــ **targetObject** قد تمت معالجته كجزء من عملية Search متسلسلة وبالتالي يستثني في الطلب الفرعي التالي.

الملاحظة – تكون الأسماء في **contextPrefix** أو **alreadySearched** أسماء مميزة أولية ولا تحتوى على أسماء مميزة بديلة.

5.10 تقدم عملية

تصف قيمة **OperationProgress** حالة التقدم المحرز في عملية لها DSAs عديدين تشارك في:

```
OperationProgress ::= SET {
        nameResolutionPhase     [0]     ENUMERATED {
            notStarted          (1),
            proceeding          (2),
            completed           (3) },
        nextRDNToBeResolved     [1]     INTEGER OPTIONAL }
```

يكون للمكونات المختلفة المعاني المعرفة أدناه:

أ) يدل مكون **nameResolutionPhase** على الوصول إلى مرحلة مناولة اسم **targetObject** لعملية. وحيث يدل هذا على أن استبانة اسم لها **notStarted**، فإن DSA لم يتوصل إلى الآن إلى سياق تسمية يحتوى على RDN(s) الابتدائي لاسم. وإذا كانت استبانة اسم هي **proceeding**، فإن الجزء الابتدائي من الاسم قد تم التعرف عليه، بالرغم من أن DSA المحتفظ بالشيء المستهدف لم يتم التوصل إليه. ويدل **nextRDNToBeResolved** مقدار ما تم التعرف عليه من الاسم [انظر 5.10 ب]. وإذا كانت استبانة اسم هي completed، فإن DSA المحتفظ بالشيء المستهدف قد تم التوصل إليه، ويتواصل أداء العملية نفسها.

ب) يدل **nextRDNToBeResolved** على أن DSA لأي من RDNs في اسم **targetObject** هو التالي لاستبانته. ويأخذ شكل عدد صحيح في مدى واحد لعدد RDNs في الإسم. ويكون هذا المكون محيناً فقط إذا كان مكون **nameResolutionPhase** له قيمة **proceeding**.

6.10 معلومات أثر

تحمل قيمة **TraceInformation** إلى الأمام سجل DSAs العاملين في أداء العملية. وتستخدم لاكتشاف وجود، أو تجنب، عروات قد تظهر من المعرفة غير المتسقة أو من وجود عروات مستعار في DIT.

```
TraceInformation ::= SEQUENCE OF TraceItem

TraceItem ::= SET {
        dsa                 [0]  Name,
        targetObject        [1]  Name OPTIONAL,
        operationProgress   [2]  OperationProgress }
```

يضيف كل DSA يعمل على انتشار عملية إلى آخر بنداً جديداً في طرف تتابع **TraceItem**. ويحتوى كل **TraceItem** على:

أ) اسم DSA الذي يضيف البند؛

ب) اسم **targetObject** الذي يضيف إليه DSA البند المستقبل في الطلب القادم. وتحذف هذه المعلمة إذا جاء الطلب المتسلسل من DUA (وفي هذه الحالة تكون القيمة المتضمنة هي **object** أو **baseObject** في **XOperation**) أو إذا كانت قيمته هي نفسها (الفعلية أو الضمنية) كما في **targetObject** في **ChainingArgument** للطلب الخارج.

ج) **operationProgress** الذي يضيف إليه DSA البند المستقبل في الطلب القادم.

يكون **dsa** الاسم المميز الأولى ولا يحتوى على أسماء مميزة بديلة. ويكون كل RDN في **targetObject** تمت معالجته هو RDN الأولى. وقد تتضمن قيم مميزة بديلة مع سياقات في مكون **valuesWithContext** لـــ **AttributeTypeAndDistinguishedValue** في RDN.

7.10 نمط مرجع

تدل قيمة ReferenceType على واحد من أنواع مختلفة للمراجع المعرفة في ISO/IEC 9594-2 | ITU-T X.501.

```
ReferenceType ::= ENUMERATED {
        superior                (1),
        subordinate             (2),
        cross                   (3),
        nonSpecificSubordinate  (4),
        supplier                (5),
        master                  (6),
        immediateSuperior       (7),
        self                    (8),
        ditBridge               (9) }
```

8.10 معلومات نقطة نفاذ

توجد ثلاثة أنماط لنقاط النفاذ:

أ) تعرف قيمة **AccessPoint** نقطة معينة يحدث عندها نفاذ إلى الدليل، وعلى وجه التحديد إلى خدوم DSA أو LDAP. وعند الإشارة إلى DSA، يكون لنقطة النفاذ **Name** أي لـــ DSA المعني، وقد يكون له **PresentationAddress** يستخدم في اتصالات OSI أو IDM بـــ DSA ذلك، وفي هذه الحالة لا يكون labeledURI محيناً. وعند الإشارة إلى خدوم LDAP، يكون لنقطة النفاذ **LabeledURI** لتستعمل في اتصالات LDAP بخدوم LDAP ذلك. وعندما يكون بند **LabeledURI** محيناً، يتم تجاهل **Name** و**PresentationAddress**، ولا يحين **SET OF protocolInformation**.

```
AccessPoint ::= SET {
    ae-title              [0]   Name,
    address               [1]   PresentationAddress,
    protocolInformation   [2]   SET SIZE (1..MAX) OF ProtocolInformation OPTIONAL,
    labeledURI            [6]   LabeledURI OPTIONAL }

LabeledURI ::= DirectoryString{ub-labeledURI}
```

ب) تعرف قيمة **MasterOrShadowAccessPoint** نقطة نفاذ إلى الدليل. وتعتمد **category**، سواء **master** أو **shadow**، لنقطة النفاذ على ما إذا كانت تشير إلى سياق اسم أو منطقة مستنسخة قابلة للاستخدام عامة. ويدل مكون **chainingRequired** على ما إذا كانت السلسلة مطلوبة لـــ DSA ذلك، أي، لا يعاد مرجع DSA ذلك.

```
MasterOrShadowAccessPoint ::= SET {
    COMPONENTS OF             AccessPoint,
    category            [3]   ENUMERATED {
        master                    (0),
        shadow                    (1) } DEFAULT master,
    chainingRequired [5]  BOOLEAN DEFAULT FALSE }
```

ج) تعرف قيمة **MasterAndShadowAccessPoints** مجموعة نقاط نفاذ إلى الدليل، أي، مجموعة متعلقة بخدومي DSAs و/أو LDAP. وتتقاسم نقاط النفاذ هذه الملكية التي تشير إليها كل نقطة إلى خدوم DSA أو LDAP المحتفظ بمعلومات مدخل من سياق تسمية مشترك (أو مجموعة مشتركة لسياقات تسمية موجودة في DSA واحد عندما تكون القيمة هي قيمة نعت **nonSpecificKnowledge**). وتدل قيمة **MasterAndShadowAccessPoints** على **category** كل قيمة **AccessPoint** تحتوي عليها. وليست هناك حاجة لتضمين نقطة النفاذ لخدوم DSA أو LDAP لسياق تسمية في المجموعة.

الملاحظة – ينبغي أن يتعرف التنفيذ على أن من الممكن لخدوم LDAP، حتى إذا عرف على أنه **shadow**، تحيين مداخل في الاستجابة لعملية تحيين LDAP التي تستقبلها.

```
MasterAndShadowAccessPoints ::= SET SIZE (1..MAX) OF MasterOrShadowAccessPoint
```

تعرف قيمة AccessPointInformation نقطة نفاذ واحدة أو أكثر إلى الدليل.

```
AccessPointInformation ::= SET {
    COMPONENTS OF             MasterOrShadowAccessPoint,
    additionalPoints    [4]   MasterAndShadowAccessPoints OPTIONAL }
```

وفي حالة الطبعة الأولى لـــ DSAs منتجين لقيمة **AccessPointInformation**، المكون الاختياري للمجموعة غائباً. وفي حالة الطبعة الأولى لـــ DSAs مفسر قيمة **AccessPointInformation**، يتم تجاهل أي قيمة لـــ **MasterAndShadowAccessPoints**.

في حالة الطبعة الثانية والتالية لـــ DSAs، قد يكون مكون قيمة **MasterOrShadowAccessPoint** المنتج لأجل قيمة **AccessPointInformation** من فئة master أو shadow، كما تحدد من قبل إجراء اختيار المعرفة لـــ DSA المنتج للقيمة. ويمكن النظر إليه كنقطة نفاذ مقترحة يوفرها DSA المولد لقيمة DSA المستقبل لها. ويمكن إنتاج أيضاً قيمة **MasterAndShadowAccessPoints** اختيارياً من أجل قيمة **AccessPointInformation**. ويشكل هذا معلومات إضافية يمكن استخدامها من قبل إجراء اختيار معرفة DSA لتحديد نقطة نفاذ بديلة.

9.10 معرفة جسر DIT

تعرف قيمة **ditBridgeKnowledge** نقطة معينة تنفذ إلى DIT آخر، وبالتحديد إلى خدوم DSA أو LDAP. وتحدد **accessPoint ditBridgeKnowledge** التي يمكن عندها لخدوم DSA أو LDAP أن ينفذ.

```
DitBridgeKnowledge  ::=  SEQUENCE {
        domainLocalID                   DirectoryString{ub-domainLocalID} OPTIONAL,
        accessPoints                    MasterAndShadowAccessPoints }
```

يتضمن المجال domainLocalID وصفاً سهل القراءة يعرّف DIT المدرج في المرجع.

10.10 الاستبعاد

كما عرف في 3.10، يستخدم مكون **exclusions** لـــ **ChainingArguments** لحد مدى عملية Search بواسطة تحديد عدد من مداخل تابعة لشيء مستهدف لا يتضمن، مع جميع توابعه، في معالجة عملية Search. ويعرف مكون **exclusion** كقيمة لنمط **Exclusions** ASN.1.

```
Exclusions ::= SET SIZE (1..MAX) OF RDNSequence
```

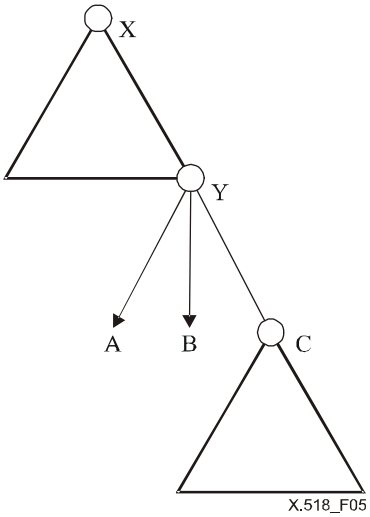
ينبغي أن تحدد كل قيمة **RDNSequence** في مجموعة **Exclusions** سابقة السياق لسياق تسمية تابع لشيء مستهدف. وإذا استقبل DSA طلب search مع قيمة **RDNSequence** لا تتوافق مع هذا القيد، قد يتجاهل DSA تلك القيمة. و**RDNSequence** هو نسبي لشيء مستهدف، وليس اسماً مميزاً لسابقة السياق.

تكون **Exclusions** هي الأسماء المميزة الأولية. ويمكن أن تتضمن أسماء مميزة بديلة ومعلومات سياق.

يمكن استخدام **Exclusions**، بجانب أنها جزء من طلب مستعمل، من قبل DSAs للحد من استنساخ معلومات معادة من طلبات فرعية لـــ Search تؤدى في وجود معلومات مظللة.

يوضح الشكل 5 مثالاً لاستخدام **Exclusions**. وفي هذا المثال، يحتفظ DSA بمنطقتين مستنسختين، واحدة تحت الأخرى. وتبدأ واحدة مع سابقة سياق X، والأخرى مع سابقة سياق C. ويكون لنسخة مدخل عند Y ثلاثة مراجع تابعة لسياقات تسمية A وB وC.

وإذا، كمثال، تمت تأدية شجرة فرعية Search في DSA هذا، إبتداء مع شيء قاعدة في سياق تسمية X، يمكن أن يوفر DSA معلومات من المنطقتين المستنسختين X وC. ويتعين توفير المعلومات من سياقات تسمية A وB عبر مراجع تابعة. وعند أداء تجزئة طلب، تحدد مراجع الاستمرارية، المستخدمة إما في **partialResults** أو سلسلة، Y كشيء مستهدف وC كعنصر وحيد لمجموعة **Exclusions**.



الشكل 5 – الاستبعاد

11.10 مراجع الإستمرارية

تصف **ContinuationReference** كيفية استمرار أداء كل أو جزء من عملية على خدوم DSA أو LDAP مختلف أو تركيب منهما. ويعاد كمرجع عندما لا يتمكن DSA العامل أو يكون غير مستعد لانتشار الطلب نفسه.

```
ContinuationReference ::= SET {
    targetObject          [0]  Name,
    aliasedRDNs           [1]  INTEGER OPTIONAL, -- only present in first edition systems
    operationProgress     [2]  OperationProgress,
    rdnsResolved          [3]  INTEGER OPTIONAL,
    referenceType         [4]  ReferenceType,
    accessPoints          [5]  SET OF AccessPointInformation,
    entryOnly             [6]  BOOLEAN DEFAULT FALSE,
    exclusions            [7]  Exclusions OPTIONAL,
    returnToDUA           [8]  BOOLEAN DEFAULT FALSE,
    nameResolveOnMaster   [9]  BOOLEAN DEFAULT FALSE }
```

يكون للمكونات المختلفة المعاني المعرفة أدناه:

أ) يدل مكون **targetObject** على الاسم المقترح لاستعماله في استمرار عملية. وقد يكون هذا مختلفاً عن الاسم المستقبل في **targetObject** لطلب قادم إذا، مثلاً، تم إبدال مستعار أو تم تحديد موقع شيء قاعدة في بحث.

تكون RDNs في **targetObject** RDNs الأولية (لأن RDNs قد تم معالجتها فعلاً). وقد تتضمن قيم مميزة بديلة مع السياق.

ب) يدل مكون **aliasedRDNs** على عدد RDNs (إن وجد) في اسم شيء مستهدف أنتجه إبدال مستعار. ويكون المتغير محيناً إذا تم إبدال مستعار.

الملاحظة – يتوفر هذا المكون من أجل الملاءمة مع تنفيذ الطبعة الأولى من الدليل. ويلغي دائماً DUAs (و DSAs) المنفذين طبقاً لآخر طبعات لمواصفات الدليل المعلمة هذه من **CommonArguments** لطلب تالي. وبهذه الطريقة، لا يشور الدليل خطأ إذا بدل مستعارون بمستعارين آخرين.

ج) يدل **operationProgress** على مقدار استبانة اسم تم تحقيقه، والذي يحكم الأداء الاضافي للعملية بواسطة DSAs مسميين، إذا رغب DSA أو DUA المستقبل لـــ **ContinuationReference** متابعتها.

د) تدل قيمة مكون **rdnsResolved** (الذي يحتاج إلى تحيين فقط إذا كانت بعض RDNs في اسم لا تخضع لاستبانة اسم كامل، ولكن افترض أن تكون صحيحة من إسناد مرجعي) على عدد RDNs تم استبانتها فعلاً، باستخدام مراجع داخلية فقط.

هـ) يدل مكون **referenceType** على أي نمط من المعرفة قد استخدم لتوليد هذه الاستمرارية.

و) يدل مكون **accessPoints** على نقاط نفاذ التي يتم الاتصال بها لتحقيق هذه الاستمرارية. وفقط عندما تعمل مراجع تابعة غير محددة يمكن أن يكون هناك أكثر من بند واحد لـــ **AccessPointInformation**.

ز) يضبط مكون **entryOnly** على **TRUE** إذا كانت العملية الأصلية بحث، مع متغير **subset** مضبوط على **oneLevel**، وتمت مواجهة مدخل مستعار كتابع مباشر لـــ **baseObject**. ويؤدى DSA الذي يؤدى بنجاح استبانة اسم على اسم **targetObject** تقييم شيء على مدخل مسمي فقط.

ح) يعرف مكون **exclusions** مجموعة من سياقات تسمية تابعة لا ينبغي أن يستكشفها DSA المستقبل.

ط) يورد عنصر **returnToDUA** اختيارياً عندما يرغب DSA الذي يولد مرجع الاستمرارية أن يدل على أنه غير مستعد لإعادة معلومات عبر DSA وسيط (مثلاً، من أجل أسباب أمنية)، ويرغب في أن يدل على أن المعلومات قد تتاح مباشرة عبر عملية عبر DAP أو LDAP بين DUA أو زبون LDAP وDSA. وعندما يضبط **returnToDUA** على **TRUE**، يمكن ضبط **referenceType** على **self**.

ي) يورد عنصر **nameResolveOnMaster** اختيارياً عندما يواجه DSA الذي يولد مرجع الإستمرارية NSSRs. وإذا ضبط على **TRUE**، يشور أن استبانة اسم متتالي، أي، مواءمة RDNs المتبقية من **nextRDNToBeResolved**، لا تستخدم معلومات نسخٌ مدخل، بما في ذلك نسخاً يمكن كتابتها في تنفيذ متعدد الرؤساء؛ وتتم الإستبانة التالية لكل RDN متبقي في DSA الرئيسي لمدخل عرفه RDN ذلك (انظر 1.20).

11 الربط وفك الربط

يستخدم **DSABind** و**DSAUnbind**، على التوالي، من قبل DSA في بداية ونهاية فترة النفاذ إلى DSA آخر. ولا يسبب الربط أو فك الربط لتصاحب DSP، في حد ذاته، خسارة لأي نتائج متصفحة موزعة قد طلبت خلال التصاحب.

1.11 ربط DSA

تستخدم عملية **DSABind** لبدء فترة التعاون بين DSA اثنين يوفران خدمة الدليل.

```
DSABind ::= BIND
        ARGUMENT        DirectoryBindArgument
        RESULT          DirectoryBindResult
        BIND-ERROR      DirectoryBindError
```

إن مكونات **DSABind** مماثلة لنظرائها في DirectoryBind (انظر ITU-T X511/ISO/IEC 9594-3) مع الاختلافات التالية:

- إن **Credentials** لـــ **DirectoryBindArgument** تسمح للمعلومات بالتعرف على AE-Title لـــ DSA الابتدائي لإرساله إلى DSA المستجيب. ويكون AE-Title في شكل اسم مميز للدليل.
- إن **Credentials** لـــ **DirectoryBindResult** تسمح للمعلومات بالتعرف على AE-Title لـــ DSA المستجيب لإرساله إلى DSA الابتدائي. ويكون AE-Title في شكل اسم مميز.
- يمكن لـــ DSA أو AE-Title استخدام أسماء مميزة بديلة ويمكن أن تتضمن معلومات سياق.

الملاحظة 1 – عند استخدام أسماء في تفويضات بسيطة أو قوية، من الممكن استخدام أسماء مميزة بديلة، إذا وجدت. ومع ذلك، قد لا يعمل الاستيقان ومراقبة النفاذ القائمين على الإسم كما يراد إذا لم يستخدم الإسم المميز الأولى. وعقب المعالجة الناجحة لعملية BIND مستيقنة، بغض النظر عن الاسم المستخدم في متغير BIND، تعرف الكيانات الموثقة كل منها الآخر بواسطة أسمائها المميزة الأولية، لتيسير عملية مراقبة النفاذ بينما BIND ساري المفعول.

الملاحظة 2 – يمكن أن يحمل عنصر خدمة تبادل الأمن التفويضات المطلوبة للاستيقان (انظر ISO/IEC 9594-5 | ITU-T X.519) وفي هذه الحالة لا توجد في متغيرات الربط أو النتائج.

2.11 فك ربط DSA

إن فك الربط في نهاية فترة التعاون بين DSA اثنين لتوفير خدمة الدليل هو من أجل بيئة OSI المحددة في 4.6.7 و5.6.7 من ISO/IEC 9594-5 | ITU-T X.519 ومن أجل بيئة TCP/IP في 2.3.9 من ISO/IEC 9594-5 | ITU-T X.519.

12 عمليات متسلسلة

لكل عملية من العمليات المستخدمة للنفاذ إلى خدمة مجردة للدليل، هناك عملية تستخدم بين DSAs متعاونين في توافق واحد مع الآخر. واختيرت أسماء العمليات لتعكس التوافق باستخدام سابقات لأسماء العمليات المختلفة المستخدمة بين DSAs المتعاونين مع مصطلح "متسلسلة".

تشكل متغيرات ونتائج وأخطاء عمليات متسلسلة، باستثناء واحد، على نحو منتظم من متغيرات ونتائج وأخطاء عمليات متوافقة في خدمة مجردة للدليل (كما ورد في 1.12). والاستثناء الوحيد عملية **ChainedAbandon**، التي تماثل بشكل منتظم نظير خدمة دليلها (كما ورد في 2.12).

1.12 عمليات متسلسلة

يمكن لـــ DSA، مستقبل لعملية من DUA أو زبون LDAP، اختيار أن يركب شكل متسلسل لتلك العملية لانتشار DSA آخر. ويمكن لـــ DSA، المستقبل لشكل متسلسل لعملية، أن يختار أيضاً تسلسله مع DSA آخر. ويمكن لـــ DSA ينفذ شكلاً متسلسلاً لعملية أن يوقع أو يجفر، أو يوقع ويجفر متغير العملية؛ ويمكن لـــ DSA المؤدى للعملية، إذا طلب ذلك، أن يوقع أو يجفر، أو يوقع ويجفر النتيجة أو الخطأ الذي يعيده المستجيب للعملية. ويمكن لـــ DSA، المستقبل لعملية من زبون LDAP أو مستقبل لعملية LDAP من DSA آخر، أن يختار انتشار العملية التي وردها زبون LDAP الأصلي إلى خدوم LDAP.

ويحدد الشكل المتسلسل لعملية باستخدام نمط له معلمات **{ } chained**.

```
chained { OPERATION : operation } OPERATION ::= {
        ARGUMENT  OPTIONALLY-PROTECTED {
            SET {
                chainedArgument       ChainingArguments,
                argument          [0] operation.&ArgumentType } }
        RESULT      OPTIONALLY-PROTECTED {
            SET {
                chainedResult         ChainingResults,
                result            [0] operation.&ResultType } }
        ERRORS{ operation.&Errors EXCEPT referral | dsaReferral }
        CODE    operation.&operationCode }
```

الملاحظة 1 – يمكن لعمليات خدمة مجردة للدليل التي تستخدم معلمة فعلية لـــ **{ } chained**.أن تشمل خطأ **abandoned**. ويعكس وجود هذا الخطأ فيما بين مجموعة من الأخطاء الممكنة لعملية متسلسلة الإمكانية التي تمت مناقشتها في 2.12، بأن **chainedAbandon** يمكن أن تولد لعملية **chainedModify** عندما يفشل تصاحب مترابط.

الملاحظة 2 – تنطبق المواصفة النهائية لخدمة مجردة لـــ DSA في الملحق A على نمط له معلمات لتركيب جميع العمليات المتسلسة لخدمة مجردة.

ويكون لمتغير العملية المشتقة المكونات:

أ) **chainedArgument** – هذه قيمة **ChainingArguments** التي تحتوى على تلك المعلومات، علاوة على المتغير الذي ورده DUA الأصلي أو زبون LDAP، المطلوب لأداء خدوم DSA أو LDAP لتنفيذ العملية. ويعرف نمط هذه المعلومات في 3.10.

ب) **argument** – هذه هي قيمة **operation.&Argument** وتتألف من متغير ورده DUA الأصلي، كما حدد في القسم المناسب من ISO/IEC 9594-3 | ITU-T X.511، أو متغير ورده LDAP الأصلي، كما حدد في القسم المناسب من RFC 2251.

الملاحظة 3 – من الممكن أيضاً تغليف أنماط PDU غير الناشئة من DAP أو LDAP إذا اعتبرت ملائمة. وتركت مواصفة الآليات التي تفعل ذلك لمزيد من الدراسة.

إذا نجح الطلب، يكون للنتيجة المشتقة من العملية المكونات:

أ) **chainedResult** – هذه قيمة **ChainingResults** التي تحتوى على تلك المعلومات، علاوة على الموردة إلى DUA الأصلي، التي قد يحتاجها DSAs سابقين في سلسلة. ويعرف نمط هذه المعلومات في 4.10.

ب) **result** – هذه قيمة **operation.&Result** وتتألف من النتيجة التي يعيدها منفذ هذه العملية، والتي القصد منها أن تعاد في النتيجة إلى DUA الأصلي. وحددت هذه المعلومات في القسم المناسب من ISO/IEC 9594-3 | ITU-T X.511.

إذا فشل الطلب، يعاد خطأ من أخطاء مجموعة **operation.&Result**، باستثناء أن **dsaReferral** يعاد بدلاً من **referral**. ومجموعة الأخطاء، التي يمكن الإبلاغ عنها، هي كما وردت لعمليات متوافقة في ISO/IEC 9594-3 | ITU-T X.511. ويرد وصف خطأ **dsaReferral** في 2.13.

2.12 عملية ترك متسلسلة

تستخدم عملية **chainedAbandon** من قبل DSA واحد ليدل لآخر أنه لم يعد مهتما بالتنفيذ السابق لعملية تم أداؤها. وقد يكون هذا لعدد من الأسباب، منها الأمثلة التالية:

- العملية التي أدت بـــ DSA المتسلسل الأصلي بترك نفسها، أو أجهضت ضمنيا بواسطة توقف التصاحب؛
- حصل DSA على المعلومات الضرورية بطريقة أخرى، مثلاً، من DSA مستجيب أسرع عامل في سلسلة متعددة موازية.

لا يضطر DSA أبداً إلى إصدار **chainedAbandon** أو لترك عملية فعلياً إذا طلب منه ذلك.

إذا نجح **chainedAbandon** فعلياً في وقف أداء عملية، فتعاد نتيجة وتعيد عملية شيء خطأ **abandoned**. وإذا لم ينجح **chainedAbandon** في وقف العملية، تعيد نفسها خطأ **abandonFailed**.

3.12 عمليات متسلسلة وصيغة بروتوكول

إن العمليات التي تتطلب صيغة بروتوكول أكبر من v1 (مثل عملية **modifyEntry** مع متغيرات معينة) أو التي تعيد نتائج مختلفة عندما تستخدم مع صيغة بروتوكول أكبر من v1 (مثل **modifyEntry** مع متغير موقع) تسلسل فقط على تصاحب مع نفس الصيغة أو رقم صيغة أكبر من المستخدم لنقل الطلب.

13 أخطاء متسلسلة

1.13 مقدمة

في معظم الأحوال، يمكن إعادة نفس الأخطاء في الخدمة المجردة لـــ DSA التي يمكن إعادتها في خدمة مجردة للدليل. والاستثناءات هي أن "خطأ" **dsaReferral** يعاد (انظر 2.13)، بدلاً من **Referral**، ويكون لمشاكل الخدمة التالية نفس قواعد تركيب مجردة ولكن علم دلالات مختلف:

أ) **invalidReference** – اكتشف DSA المعيد لهذا الخطأ خطأ في تسمية معرفة DSA كما حدد في متغير سلسلة **referenceType**.

ب) **loopDetected** – اكتشف DSA المعيد لهذا الخطأ عروة في معلومات المعرفة في الدليل.

إن إجراء الأخطاء التي يمكن أن تحث هو نفسه لإجراء خدمة مجردة للدليل، كما ورد في ISO/IEC 9594-3 | ITU-T X.511.

إذا حدث خطأ خلال عملية تسلسل، يمكن أن يوقع DSA المستجيب أو يجفر أو يوقع ويجفر الخطأ المعاد.

2.13 مرجع DSA

يولد DSA خطأ **dsaReferral** عندما لا يرغب، لسبب من الأسباب، في استمرار أداء عملية بواسطة تسلسل العملية في واحد أو أكثر من DSA. وترد في 3.8 الظروف التي يمكن فيها إعادة مرجع.

```
dsaReferral   ERROR ::= {
        PARAMETER OPTIONALLY-PROTECTED {
            SET {
                reference           [0]   ContinuationReference,
                contextPrefix       [1]   DistinguishedName OPTIONAL,
                COMPONENTS OF CommonResults } }
        CODE            id-errcode-dsaReferral }
```

يكون للمعلمات المختلفة المعاني كما يرد أدناه:

أ) يحتوي **ContinuationReference** على المعلومات المطلوبة من المنفذ لانتشار طلب مناسب إضافي، ربما لـــ DSA آخر أو خدوم LDAP. ويرد نمط هذه المعلومات في 11.10.

ب) إذا كان مكون **returnCrossRefs** لـــ **ChainingArguments** لهذه العملية القيمة **TRUE**، ويقوم المرجع على أساس تابع أو إسناد مرجعي، فإن معلمة **contextPrefix** قد تتضمن اختيارياً. وتقرر السلطة الإدارية لأي DSA أي مراجع معرفة، إن وجدت، يمكن إعادتها بهذه الطريقة، مثلاً، قد تكون سرية لـــ DSA ذلك).

يكون **contextPrefix** أو مرجع إستمرارية الاسم المميز الأولى. وقد تتضمن قيماً مميزة بديلة مع سياق في مكون **valuesWithContext** لـــ **AttributeTypeAndDistinguishedValue** لأي RDN.

ويمكن أن تؤهل المعلومات الموفرة باستخدام مكون **notification** لـــ **CommonResults**.

القسم 5 – الإجراءات الموزعة

14 مقدمة

1.14 المدى والحدود

يحدد هذا القسم الإجراءات لعملية موزعة للدليل يؤديها DSAs. ويؤدى كل DSA منفرد الإجراءات الواردة أدناه؛ ويؤدى العمل الجماعي لجميع DSAs إلى مجموعة كاملة من الخدمات يوفرها الدليل للمستعملين.

2.14 المطابقة

يقوم وصف إجراءات DSA في هذا القسم على أساس النماذج في القسمين 8 و9 من ISO/IEC 9594-2 | ITU-T X.501 وفي القسمين 7 و8 من مواصفة الدليل هذا. وتعتبر المخططات الانسيابية ووصفها النصي المتطابق إحدي وسائل تقابل مجموعة ما من مدخلات خارجية (DAP وLDAP و/أو DSP) بـــ DSA في واحد أو أكثر من دخل خارجي (أي، نتيجة أو خطأ أو مرجع أو طلب متسلسل) أنتجه DSA ذلك، معتمداً على شجرة معلومات DSA معينة يحتفظ بها DSA ذلك.

ومن المحتمل أن يوزع الدليل عبر DSAs منفذين طبقاً لطبعات مختلفة من مواصفات الدليل، وكذلك DSAs المنفذين لدعم LDAP فقط. ولا يكون DUA أو زبون LDAP البادئ للطلب على وعي بأي طبعة يقوم DSA أو DSAs المبين لطلب زبون DUA أو LDAP بتنفيذها. ولهذا، للسماح لعملية في بيئة متغايرة الخواص، ينفذ DSA طبقاً لقواعد التمديد المعرفة في القسم 12 من ISO/IEC 9594-5 | ITU-T X.519.

الملاحظة 1 – إن DSAs المنفذين لدعم LDAP فقط قد ينفذون أو لا ينفذون طبقا لقواعد التمديد.

إن تنفيذ DSA يكون مماثلاً وظيفياً للسلوك الخارجي الذي تحدده الإجراءات الواردة هنا. إن الخوارزمية المستخدمة من قبل تنفيذ DSA معين لاشتقاق خرج (مخرجات) من مدخلات ما وشجرة معلومات DSA المحتفظ بها ليست معيارية.

الملاحظة 2 – إن القصد من المخططات الانسيابية التي تصاحب الإجراءات هو استخدامها كمعينات نحو فهم الإجراءات. ولا تعتبر على أنها بديل دقيق للوصف النصي. وعندما يكون هناك تفاوت بين الوصف النصي ومخطط انسيابي لإجراء معين، فالقصد هو أن يكون السبق للوصف النصي.

1.2.14 التفاعل المتضمن أول طبعة لـــ DSA

إذا قيمت عمليات تعديل عبر حدود DSA (أي، **addEntry** مع **TargetSystem** أو Remove أو Rename سابقة سياق)، فإن مواصفة الدليل هذا تحدد فقط سلوك DSA اثنين للطبعة الثانية أو طبعة لاحقة. إن التفاعل بين طبعتين أوليين لـــ DSAs أو بين الطبعة الأولى لـــ DSA والطبعة الثانية أو اللاحقة لـــ DSA هي خارج مدى مواصفات الدليل. وعندما يكون لطبعة مختلطة لـــ DSAs إسناد تشغيلي تراتبي، قد تسمح معرفة كل طبعة لأخرى بخطأ متسق يعطى للمستعمل.

3.14 نموذج مفهومي

يظهر تعقيد عملية موزعة للدليل الحاجة إلى وضع نماذج مفهومية باستخدام كل من التقنيات الوصفية السردية والصورية. ومع ذلك، لا ينبغي تركيب مخططات سردية أو تصويرية كوصف رسمي لعملية دليل موزعة.

4.14 عملية فردية وتعاونية لـــ DSAs

ينظر النموذج إلى عملية DSA من منظورين منفصلين، عندما يؤخذان معاً، يوفران صورة كاملة تشغيلية للدليل.

أ) **DSA-centered perspective** – في هذا المنظور، توصف مجموعة الإجراءات التي تدعم الدليل من وجهة نظر DSA وحيد. ويجعل هذا من الممكن توفير مواصفة نهائية لكل إجراء ويأخذ في الحسبان الكامل علاقاتها المتبادلة وهيكل الرقابة الشامل. وتصف الأقسام من 16 إلى 22 إجراءات DSA من منظور DSA ممركز.

ب) **operation-centered perspective** – توفر نظرة DSA ممركز تفاصيل كاملة ولكن تجعل من الصعب فهم هيكل العمليات الفردية، التي قد تمر بمعالجة بواسطة DSAs متعددين. وبناءً على ذلك، يعتمد القسم 15 نظرة أولية لعملية ممركزة لتقديم مراحل المعالجة المطبقة على كل منها.

ولدعم العملية الموزعة للدليل، يؤدى كل DSA أعمال مطلوبة لتحقيق القصد من كل عملية وأعمال اضافية مطلوبة لتوزيع ذلك التحقق عبر DSAs متعددين. ويستكشف القسم 15 التمييز بين هذين النوعين من الأعمال. وفي الأقسام من 16 إلى 22، يرد كل من نوعي الأعمال بالتفصيل.

5.14 الاتفاقات التعاونية بين DSAs

إن جميع DSAs الذين في علاقة تابعة/عليا نتيجة لسياقات تسمية يحتفظون بها، لديهم إسنادات تراتبية و/أو تشغيلية تراتبية غير محددة بينهم، يعتمد ذلك على أنماط مرجع معرفة يحتفظ به DSAs.

يمكن إدارة إسنادات تراتبية وتشغيلية تراتبية غير محددة بين DSAs باستخدام الإجراءات الواردة في القسمين 24 و25 أو بواسطة وسائل أخرى (مثل، الهاتف).

إن DSA المحتفظ بمداخل في منطقة إدارية لـــ DSA أعلى يدير تخطيط فرعي، ويتبع governing-search-rule (إن وجد) ويراقب النفاذ إلى مداخل كما تطلب السلطة الإدارية. ويمكن أداء تنظيم مداخل في منطقة إدارية كما عرف في ITU-T X501/ISO/IEC 9594-2 أو بواسطة آليات محلية.

15 سلوك دليل موزع

1.15 التنفيذ التعاوني لعمليات

يجهز كل DSA بإجراءات قادرة على التنفيذ الكامل لجميع عمليات الدليل. وفي حالة احتواء DSA على DIB كاملة، تنفذ جميع العمليات، في الحقيقة، بالكامل في DSA ذلك. وفي حالة أن DIB موزعة عبر DSAs متعددين، يتجزأ اكتمال عملية نمطية، مع جزء من تلك العملية منفذ في كل من DSAs المتعاونين المحتملين الكثيرين.

وفي البيئة الموزعة، يرى DSA نمطي كل عملية كحدث عابر: ينفذ العملية DUA أو زبون LDAP أو DSA آخر؛ وينفذ DSA المعالجة على شيء ثم يوجهه نحو DSA آخر لمزيد من المعالجة.

والنظرة البديلة هي اعتبار المعالجة الكلية التي تمر بها عملية خلال تنفيذها بواسطة DSAs متعاونين. ويكشف هذا المنظور مراحل المعالجة المشتركة التي تنطبق على جميع العمليات.

2.15 مراحل معالجة عملية

ينظر إلى كل عملية للدليل على أنها تتألف من ثلاث مراحل متميزة:

أ) مرحلة *Name Resolution* حيث يكون الاسم فيها هو شيء يكون مدخله عملية خاصة تؤدى تستخدم لتحديد موقع DSA الذي يحتفظ بالمدخل؛

ب) *Evaluation phase* تحدد فيها عملية بواسطة طلب دليل معين (مثل عملية Read) يجرى بالفعل أداؤها؛

ج) *Results Merging phase* التي تعاد فيها نتائج عملية محددة إلى DUA أو زبون LDAP الطالب. وإذا اختير أسلوب سلسلة للتفاعل، قد تتضمن Results Merging phase العديد من DSAs، يتسلسل كل واحد

بالطلب الأصلي أو الطلب الفرعي (كما عرف في 1.3.15 – تجزئة طلب) بـــ DSA آخر خلال أي أو كلا المراحل السابقة.

وفي حالة عمليات Read وCompare وList وSearch وModify Entry وModify DN وRemve Entry، تحدث استبانة اسم على اسم شيء موفر في متغير العملية. وفي حالة Add Entry، يكون المدخل المستهدف لاستبانة اسم هو المدخل الأعلى المباشر المتوفر في متغير العملية – ويمكن اشتقاقه بسهولة بواسطة سحب RDN النهائي من الاسم الموفر في متغير العملية. (ويتم هذا عبر متغير محلي **m** في إجراء **FindDSE** في 1.3.18).

يمكن توجيه عملية على مدخل معين ابتدائياً عند أي DSA في الدليل. ويستخدم DSA ذلك معرفته، من الممكن بالتزامن مع DSAs آخرين، لمعالجة العملية من خلال ثلاث مراحل.

1.2.15 مرحلة استبانة اسم

إن استبانة اسم هي عملية مواءمة تتبعية لكل RDN في اسم مستهدف لقوس (أو ذروة) DIT، تبدأ منطقياً عند Root والتقدم إلى أسفل في DIT. ومع ذلك، وبسبب أن DIT مميزة بين DSAs كثيرين اعتباطيين، يمكن لكل DSA فقط أن يؤدى جزء من عملية استبانة اسم. ويؤدى DSA ما جزأه من عملية Name Resolution بواسطة استعراض شجرة معلومات DSA المحلية. ويرد وصف هذه العملية في القسم 18 والرسومات المصاحبه (انظر الأشكال من 9 إلى 12). وعلى أساس شجرة معلومات DSA المحلية، ومعلومات المعرفة الواردة فيها، يتمكن DSA من استدلال ما إذا كان من الممكن مواصلة الاستبانة بواسطة DSA واحد أو أكثر، أو إذا كان الاسم خطأ.

وتقيد مرحلة Name Resolution العمل في DSA Information Tree إذا ضبط خيار مراقبة خدمة **manageDSAIT**.

2.2.15 مرحلة التقييم

عند اكتمال مرحلة Name Resolution، تؤدى العملية الفعلية المطلوبة (مثل، Read أو Search).

يمكن تنفيذ عمليات تتضمن استفسار مدخل وحيد – Read وCompare – بالكامل في DSA الذي يوجد فيه المدخل.

تحتاج العمليات التي تتضمن استفسار مداخل متعددة – List وSearch – إلى تحديد موقع توابع الهدف، التي توجد أو لا توجد في نفس DSA. وإذا لم يوجد جميعها في نفس DSA، تحتاج العمليات إلى توجيهها إلى DSAs محددين في تابع أو تابع غير محدد أو مورد أو مراجع رئيسية (حسب الاقتضاء) لاكمال عملية التقييم.

وتقيد مرحلة Name Resolution العمل في DSA Information Tree إذا ضبط خيار مراقبة خدمة **manageDSAIT**. وبالمثل، إذا بدأت مرحلة التقييم في خدمة محددة لمنطقة إدارية، يتقيد التقييم على المنطقة الإدارية تلك.

3.2.15 مرحلة اندماج النتائج

تدخل مرحلة Results Merging بمجرد توافر بعض نتائج مرحلة Evaluation.

في الحالات حيث تؤثر العملية على مدخل وحيد، يمكن إعادة نتيجة العملية ببساطة إلى DUA الطالب أو زبون LDAP. وفي الحالات حيث أثرت العملية على مداخل متعددة على DSAs متعددين، يمكن جمع النتائج. وإذا تم إجراء حماية للنتائج، لا تجمع النتائج. وينبغي إعادة النتائج إلى DUA أو زبون LDAP دون أداء اندماج.

وتشمل الاستجابات المسموح بها المعادة إلى الطالب بعد اندماج النتائج:

أ) نتيجة كاملة للعملية؛

ب) نتيجة لم تكتمل بسبب أن بعض أجزاء DIT تظل دون استكشاف (تنطبق على List وSearch فقط). ويمكن أن تشمل *نتيجة جزئية* مراجع استمرارية لأجزاء DIT التي لم تستكشف؛

ج) خطأ (يعتبر مرجع حالة خاصة)؛

د) إذا كان الطالب هو DSA، **ChainingResults**.

3.15 إدارة عمليات موزعة

تتضمن المعلومات في متغير كل عملية يمكن فيها أن يطلب من DSA أن يؤديها مشيراً إلى التقدم المحرز في كل عملية بينما تستعرض DSAs مختلفين للدليل. ويجعل هذا من الممكن لكل DSA أن يؤدى الجانب الملائم من المعالجة المطلوبة، ويسجل اكتمال ذلك الجانب قبل توجيه العملية إلى الخارج نحو DSAs آخرين.

وتتضمن إجراءات اضافية في DSA للتوزيع المادي للعمليات ولدعم احتياجات أخرى تنشأ من توزيعها.

1.3.15 تجزئة طلب

إن تجزئة طلب هي عملية يؤديها DSA داخلياً قبل الاتصال بـــ DSA واحد أو آكثر آخر وخدومي LDAP. ويجزء طلب إلى عديد من الطلبات الفرعية بحيث إن كل طلب فرعي ينجز جزءاً من المهمة الأصلية. ويمكن استخدام تجزئة طلب، مثلاً، في عملية بحث، بعد العثور على قاعدة شيء. وبعد التجزئة، يمكن تسلسل الطلبات الفرعية بشكل أحادي أو متعدد مع DSAs آخرين و/أو خدومي LDAP لمواصلة المهمة.

يكون **argument** لطلب متسلسل (انظر 1.12) أو طلب فرعي هو متغير عملية غير معدلة إذا كان DUA قد بدأها وتكون LDAPMessage غير معدلة إذا كان زبون LDAP قد بدأها. ولا يغير DSA المستقبل لطلب **argument** عند القيام بتجزئة طلب.

ملاحظة – تحدد الأقسام التالية متطلب مكونات فردية لـــ **argument**. ولا ينبغي تفسير هذا ليعني أن المكون الذي لم يذكر صراحة يمكن تغييره.

2.3.15 DSA كمستجيب لطلب

يمكن لـــ DSA مستقبل لطلب التأكد من التقدم المحرز في ذلك الطلب باستخدام معلمة **operationProgress**. ويحدد هذا ما إذا كانت العملية ما تزال في مرحلة Name Resolution أو وصلت لمرحلة التقييم، وأي جزء من العملية ينبغي على DSA أن يلبيها. وإذا لم يستطع DSA تلبية الطلب بالكامل، فإما أن يمرر العملية (بواسطة سلسلة وحيدة أو سلسلة متعددة) إلى DSA واحد أو أكثر و/أو خدومي LDAP الذين يمكنهم المساعدة في تلبية الطلب، أو إعادة مرجع إلى DSA آخر أو خدوم LDAP، أو ينهي الطلب بخطأ.

3.3.15 اكتمال عمليات

إن كل DSA بدء عملية أو عمل على انتشار عملية على DSA واحد أو أكثر و/أو خدومي LDAP يتتبع وجود تلك العملية حتى يعيد كل DSA آخر و/أو خدوم LDAP نتيجة أو خطأ، أو انقضاء الحد الزمني الأقصي للعملية. وينطبق هذا المتطلب على جميع العمليات وأساليب الانتشار ومراحل المعالجة. ويضمن الغلق المنظم للعمليات الموزعة التي تم انتشارها في الدليل.

4.15 مناولة العروات

قد تكون DIT في حالة تسبب عروات. وكمثال، يمكن أن تحدث العروات خلال استبانة اسم حيث إبدال مستعار واحد أو أكثر يعيد الاستبانة إلى نفس فرع DIT. والسبب المحتمل الآخر للعروات هو من خلال سوء تشكيل مراجع معرفة.

وفي سياق عملية دليل معينة، تحدث عروة إذا أعيدت العملية في أي وقت الحالة السابقة، حيث تعرف *حالة* بواسطة المكونات التالية:

- اسم DSA الذي يعالج العملية حالياً؛
- اسم **targetObject** كما ورد في متغير العملية؛
- **operationProgress** كما ورد في متغير العملية كما عرف في 5.10.

ولا يعني هذا أن عملية لا يمكن معالجتها مرات متعددة بواسطة DSA معين. ومع ذلك، تعني أن DSA لا يعالج نفس العملية في نفس الحالة مرات متعددة.

تراقب العروات باستخدام متغير **traceInformation** كما عرف في 6.10، الذي يسجل تتابع حالات عملية معينة مرت به. وتعرف استراتيجيتان لتحديد ما إذا كانت العروات قد حَدُثت، أو على وشك الحدوث. وهي كشف العروات وتجنب العروات، ويرد وصفهما في 1.4.15 و2.4.15 على التوالي.

إن كشف العروات إلزامي وتجنب العروات اختياري.

1.4.15 كشف العروات

عند استقبال عملية لدليل، يبدأ DSA في إقرار صلاحية العملية لضمان أن بالإمكان معالجتها. وأهم مهمة لإقرار الصلاحية هو التأكد من العروات، بواسطة تحديد الحالة الراهنة للعملية الظاهرة في تتابع حالات سابقة مسجلة في متغير **traceInformation** للعملية. وهذه الخطوة للتأكد من العروات هي كشف العروات.

2.4.15 تجنب العروات

يتطلب تجنب العروات أن يقوم DSA قبل التسيير المباشر لعملية إلى DSA آخر كجزء من إجراء سلسلة، بتحديد ما إذا كانت الحالة التتابعية للعملية (التي هي **traceItem** التي يضيفها DSA المستقبل إلى **traceInformation** عندما يستقبلها) تظهر على تتابع الحالات السابقة المسجلة في متغير **traceInformation** للعملية الأصلية الواصلة.

في حالة استقبال مراجع أو العمل عليها، لا يمكن تحقيق تجنب العروات وكشف العروات بمجرد فحص **traceInformation**. وفي هذه الحالة، يعمل كل مرة DSA على مرجع، يحتاج إلى تخزين حالة تتابعية للعملية (أي، **traceItem** الذي سيقوم باضافته DSA المستقبل عندما يستقبل الطلب) مع سجل الطلب الواصل. وقبل العمل على مرجع أو إعادته، يحتاج DSA إلى التأكد من خلال هذه القائمة، وذلك للتأكد من أن الطلب المماثل لم يرسل في السابق بينما يحاول خدمة العملية الواصلة.

5.15 اعتبارات أخرى لعملية موزعة

1.5.15 مراقبة خدمة

تحتاج مراقبة خدمة إلى اعتبار خاص في البيئة الموزعة لمعالجة العملية بالطريقة التي طلبت بها.

أ) **chainingProhibited** – يستشير DSA مراقبة الخدمة هذه عند تحديد أسلوب انتشار عملية. وإذا ضبط، فإن DSA يستخدم دائماً أسلوباً مرجعياً. ومع ذلك، إذا لم يضبط، يمكن أن يختار DSA ما إذا كان يستخدم سلسلة أو مرجعاً يعتمد ذلك على قدراته.

ب) **timeLimit** – يحتاج DSA إلى أن يأخذ في الاعتبار مراقبة الخدمة هذه لضمان أن الحد الزمني لا يتم تجاوزه في DSA ذلك. وإذا طلب DSA أداء عملية بواسطة DUA، يهتم مبدئياً بـــ **timeLimit** الذي عبر عنه DUA باعتباره الوقت المنتهي المتاح بالثواني لاكتمال العملية. وإذا طلبت سلسلة، يتضمن **timeLimit** في متغير السلسلة ليمرر إلى DSA(s) التالي. وفي هذه الحالة، تستخدم نفس قيمة الحد لكل طلب متسلسل، ويكون وقت (UTC) الذي تكتمل فيه العملية لتلبية القيد المحدد الأصلي. وعند استقبال **ChainingArguments** مع **timeLimit** المحدد، يحترم DSA المستقبل هذا الحد.

ج) **sizeLimit** – يحتاج DSA إلى أن يأخذ في الاعتبار مراقبة الخدمة هذه لضمان أن قائمة النتائج لا تتجاوز الحجم المحدد. وينقل الحد، كما ورد في المتغير المشترك للطلب الأصلي، دون تغيير حيث أن الطلب متسلسل. وإذا طلب تجزئة طلب، تتضمن نفس القيمة في المتغير ليمرر إلى DSA التالي، ويستخدم الوقت الكامل لكل طلب فرعي. وعندما تعاد النتائج، يتبين لـــ DSA الطالب نتائج متعددة ويطبق الحد على الجميع لضمان أن العدد المطلوب يعاد فقط. وإذا تم تجاوز الحد، يشار إلى هذا في الإجابة.

د) **priority** – في جميع أساليب الانتشار، يكون كل DSA مسؤولاً عن ضمان إن معالجة العمليات تتم بشكل منظم بحيث تدعم مراقبة الخدمة هذه، إن وجدت.

هـ) **localScope** – تحدد العملية على مدى محلي معرف ولا يعمل كل DSA على انتشار الطلب خارج هذا.

و) **scopeOfReferral** – إذا أعاد DSA مرجعاً أو نتيجة جزئية إلى عملية List أو Search، فإن مراجع الاستمرارية المدمجة تكون في المدى المطلوب.

تحتاج جميع مراقبات الخدمة إلى احترامها، ولا يتطلب استخدامها أي اعتبار خاص في بيئة موزعة.

2.5.15 التمديدات

إذا واجه DSA عملية ممدة في مرحلة Name Resolution لمعالجة ويقرر أنه ينبغي على العملية أن تسلسل بـــ DSA واحد أو أكثر، فيشمل غير متغير في العملية المتسلسلة لأي تمديدات محينة.

ملاحظة – قد تحدد سلطة إدارية أن من الملائم إعادة **serviceError** مع مشكلة **unwillingToPerform** إذا لم ترغب في انتشار تمديد.

إذا واجه DSA تمديداً لا يدعمه في مرحلة التقييم للمعالجة، قد نشأ امكانيتان. إذا لم يكن التمديد حرجاً، يتجاهل DSA التمديد. وإذا كان التمديد حرجاً، يعيد DSA **serviceError** مع مشكلة **unavailableCriticalExtension**. إن تمديد حرج لعملية شيء متعدد قد ينتج في كل من نتائج وأخطاء خدمة هذا التنوع. ويتجاهل DSA المدمج لهذه النتائج والأخطاء أخطاء هذه الخدمة ويستخدم مكون **unavailableCriticalExtension** لـــ **PartialOutcomeQualifier** كما ورد في ITU-T X.511 | ISO/IEC 9594-3.

3.5.15 إبدال مستعار

إن إبدال مستعار هي عملية خلق اسم شيء مستهدف جديد، بواسطة استبدال جزء من اسم مميز لمدخل مستعار لاسم شيء مستهدف أصلي بقيمة نعت **AliasedEntryName** من مدخل المستعار. ولا يتأثر اسم object في العملية بواسطة إبدال مستعار.

4.5.15 استبانة أسماء لمتغيرات سياقات

خلال مرحلة استبانة اسم، وبينما يجرى معالجة RDNs يتم خلق اسم شيء مستهدف جديد بواسطة ضمان أن كل **AttributeTypeAndDistinguishedValue** في RDN يستخدم القيمة المميزة الأولية لذلك النعت باعتبارها **value**. وبهذه الطريقة، يتقدم اسم الشيء المستهدف إلى الأمام نحو اسم مميز أولي. ويتم هذا لتوفير مناولة متسقة لاسم، ولا سيما حيث يشارك DSAs في الطبعة ما قبل الثالثة في استبانة اسم. ولا يتأثر اسم **object** في هذه العملية بهذا البديل.

5.5.15 نتائج متصفحة

عندما يشمل DUA **PagedResultsRequest** في طلب **list** أو **search** (انظر 7-9 من ITU-T X.511 | ISO/IEC 9594-3)، قد يؤدى التصفح DSA الموثق مباشرة مع DUA، ويسمي أيضاً bound DSA، أو يمكن أن يؤديه DSA الذي يحتفظ بمدخل **baseObject/object** طلب **list** أو **search** (من الممكن بعد إبدال مستعار واحد أو أكثر)، ويسمي أيضاً *initial performer*. وإذا تم أداء التصفح بواسطة DSA موثق، الذي يمكن أن يكون منفذ إبتدائي، يسمي التصفح

bound-DSA paged results. وإذا تم التصفح بواسطة منفذ إبتدائي، وكان المنفذ الإبتدائي مختلفا عن DSA الموثق، فإن التصفح يسمي *DSP paged results*.

إن DSA الذي يدعم DSP paged results يقوم:

- بدعم DSA-bound paged results؛
- بدعم DSP paged results باعتباره DSA موثق؛
- بدعم DSP paged results باعتباره منفذاً ابتدائياً؛
- بدعم المكون الفرعي **entryCount** لـــ **PartialOutcomeQualifier**.

عندما يستقبل DSA موثق طلب **List** أو **search** مع **PagedResultsRequest**، لا يكون DSA الموثق هو المنفذ الإبتدائي للطلب، وقد يختار DSA الموثق ليشمل معلمة **dspPaging** في **ChainingArguments**. وقد يقوم المنفذ الابتدائي باختيار أداء DSP paged results. ويشور هذا إلى DSA الموثق بواسطة تضمين **queryReference** في **PartialOutcomeQualifier**. وهذا هو **queryReference** المعاد إلى DUA ليستخدم في استرداد الصفحة التالية.

وإذا اختار المنفذ الابتدائي ألا يدعم DSP paged results أو لا يؤديه، قد يؤدى DSA الموثق تصفح DSA الموثق العادي.

وإذا كان DSA هو المنفذ، ولكن ليس المنفذ الابتدائي، فيتجاهل مكون **dspPaging** الممكن في **chainingArguments**، ويحترم مراقبة خدمة **sizeLimit** إن وجدت.

6.15 استيقان عمليات موزعة

قد يطلب مستعملو الدليل مع السلطات الإدارية التي توفر خدمات الدليل، بناء على تقديرهم، أن يجرى استيقان عمليات الدليل. ولأي عملية معينة للدليل، يعتمد طابع عملية الاستيقان على سياسة الأمن السارية.

وتتاح مجموعتان من إجراءات الاستيقان تمكن مجتمعة تلبية مدى من متطلبات الاستيقان. وإحدي مجموعات الإجراءات هي التي يوفرها Bind: تيسر هذا الاستيقان بين كيانين لتطبيق الدليل لأغراض إنشاء تصاحب. وتستوعب إجراءات Bind مدى من تبادلات الاستيقان من مجرد تبادل بسيط لكيانات إلى استيقان قوي.

وبالإضافة إلى استيقان كيان ند لتصاحب كما يوفره Bind، تعرف إجراءات اضافية في الدليل لتمكين استيقان عمليات فردية. وتعرف مجموعتان متميزتان لإجراءات استيقان دليل. واحدة تيسر خدمات استيقان المرسل، التي تتناول الاستيقان، بواسطة DSA، لمبادر طلب الخدمة الأصلي. وتيسر المجموعة الثانية نتائج خدمات الاستيقان التي تتناول الاستيقان، بواسطة المبادر، لأي نتائج تعاد.

ومن أجل استيقان مرسل، يعرف إجراءان، واحد قائم على أساس تبادل بسيط لكيانات، يسمي **identity based authentication**، وواحد قائم على تقنيات توقيع رقمي، يسمي **signature based authentication**. إن الإجراء الأول من هذه الإجراءات هو بدائي في طابعه نظراً لأن تبادل كيان قائم على تبادل أسماء مميزة ترسل بوضوح.

وبالنسبة لاستيقان نتائج، يعرف إجراء **results authentication** وحيد، على أساس تقنيات توقيع رقمي؛ ونتيجة للتعقيد العام لطابع نسق النتائج، لا يعرف إجراء بسيط قائم على الكيان.

يمكن أن تدعم هذه الإجراءات استيقان استجابات خطأ.

إن الخدمات الواردة أدناه يتعين اعتبار أنها تزيد التي توفرها خدمة Bind؛ ويفترض أن إجراءات Bind قد تمت بنجاح قبل استيقان عمليات دليل.

وترد في القسم 22 إجراءات لينفذها DSA عند توفير استيقان المرسل والنتائج.

16 محول العمليات

إن **Operation Dispatcher** هو إجراء التحكم الرئيسي في DSA. وهو يوجه كل عملية خلال ثلاث مراحل من معالجة طلب. ولهذ يستفيد **Operation Dispatcher** من مجموعة إجراءات تعالج بالكامل الطلب كما يبين الشكل 6.

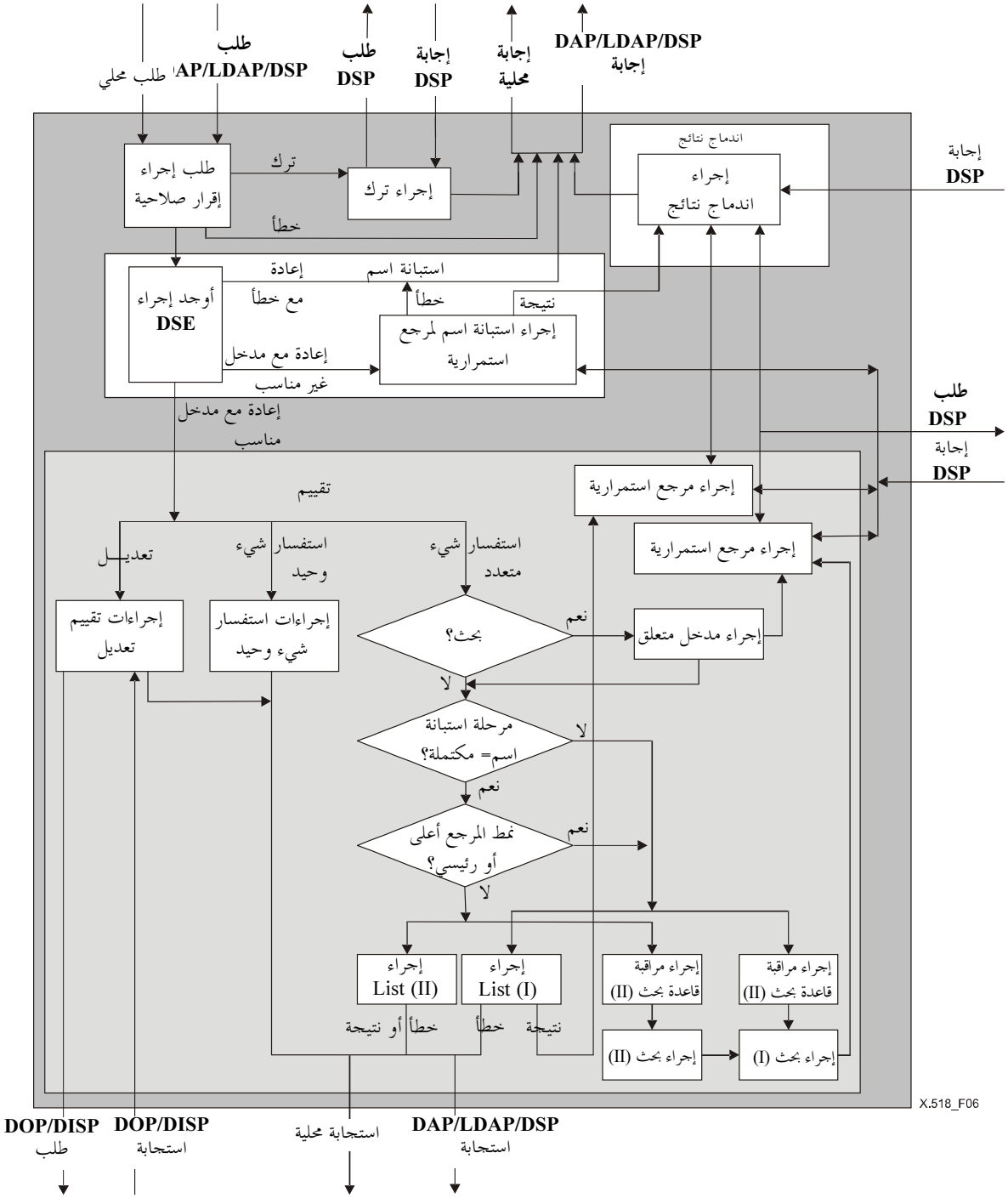
1.16 مفاهيم عامة

1.1.16 الإجراءات

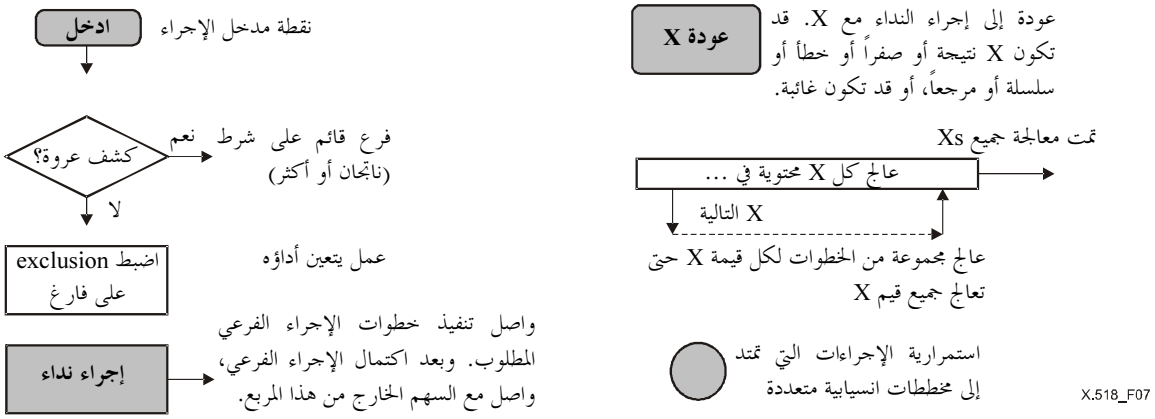
يتألف كل إجراء من الإجراءات التي يستخدمها محول العمليات من تعريف لسطح بيني مفهومي على أساس معلماته، أي، متغيرات ونتائج وأخطاء ووصف لخطوات الإجراء نفسه. ويصف سلوك الإجراءات مخططات انسيابية ونص. وفي مخطط انسيابي، يكون للرموز المستخدمة علم الدلالات التالي (انظر الشكل 7).

2.1.16 استخدام هياكل معطيات مشتركة

تستفيد جميع الإجراءات من بعض هياكل المعطيات المتاحة خلال معالجة عملية في **Operation Dispatcher**. وتقوم هياكل المعطيات هذه بخدمة تنسيق تدفق المعطيات في **Operation Dispatcher**. وتتصاحب معظم هذه الهياكل مباشرة مع متغير العملية والنتيجة التي تنشأ للعملية. ويشار إلى مكونات المتغير والنتيجة باستخدام أسمائها في تعريف ASN.1 متصاحب (مثل، مكون **operationProgress** لمتغيرات سلسلة). وإذا كان أي من هذه الهياكل هيكل مركب، يشار إلى مكون هذا الهيكل باعتباره **compound.component** (مثل، **operationProgress.nameResolutionPhase**).



الشكل 6 – محول العمليات



الشكل 7 – الرموز المستخدمة في مخططات انسيابية

تعرف هياكل المعطيات التالية في **Operation Dispatcher**:

- **NRcontinuationList** – قائمة بمراجع الاستمرارية وضعت للاستخدام في إجراء **Name Resolution Continuation Reference**.
- **SRcontinuationList** – قائمة بمراجع الاستمرارية وضعت لاستخدامها في إجراء **List** أو **Search Continuation Reference**.
- **admPoints** – قائمة بمراجع DSEs لنمط نقطة إدارية تجمع خلال Name Resolution.
- **referralRequests** – قائمة طلبات أو طلبات فرعية تم تسلسلها نتيجة لتنفيذ مراجع. ويوجز كل طلب/طلب فرعي في شكل **TraceItem**. وتستخدم هذه القائمة بواسطة إجراء Loop Avoidance في 2.4.15.
- **emptyHierarchySelect** – نمط بولاني متغير يمكن ضبطه في إجراء **Hierarchy Selection**. ويفترض ضبط المتغير عند دخول إجراء **Hierarchy Selection** لأول مرة خلال عملية Search.
- **streamedResultsOK** – نمط بولاني متغير يضبط في إجراء **Name Resolution** ليدل على نتائج متدفقة يمكن قبولها لهذه العملية. والقيمة بالتغيب لهذا المتغير هي false.

وفضلاً عن ذلك، قد يستخدم إجراء مجموعة من متغيرات معرفة محلياً.

3.1.16 أخطاء

في كل مرحلة معالجة يمكن اكتشاف خطأ خلال تنفيذ أي إجراء فرعي. ويعاد عادة كل خطأ معرف في الإجراء الفرعي هذا إلى الطالب باعتباره خطأ بروتوكول متطابق. وفي هذه الحالة، يتم إنهاء محول العملية فوراً. وفي حالة استقبال أخطاء متعددة، قد تختار الإجراءات المحلية واحداً منها ليعاد.

وبديلاً عن ذلك، قد يختار إجراء لمعالجة أخطاء (مثلاً، إذا أعيد **serviceError** مع مشكلة **busy** إلى طلب فرعي لبحث متسلسل) عند نقاط معينة لمعالجة عملية. وفي هذه الحالة، يستمر الإجراء في تنفيذه ولا يعاد خطأ إلى الطالب.

ويمكن اختارياً أن يوقع DSA أو يجفر أو يوقع ويجفر الأخطاء العائدة في عملية موزعة على أساس DirQOP المختار وكشف الأخطاء المطلوب.

4.1.16 أحداث لا تزامنية

خلال معالجة طلب عملية في Operation Dispatcher، قد تحدث أحداث لا تزامنية عديدة. وتحدد الفقرات التالية كيفية مناولة تجاوز حد الوقت أو حد الحجم أو الحد الإداري، وخسارة التصاحب وطلب Abandon لعملية تجرى معالجتها. ومناولة جميع الأحداث اللاتزامنية الأخرى، مثل، قرارات سياسة محلية وما إلى ذلك، هي خارج مواصفة الدليل هذا.

1.4.1.16 حد الوقت

يمكن انقضاء **timeLimit** كما ورد في **CommonArguments** عند أي نقطة من الزمن خلال العملية. وفي هذه الحالة، يعاد عادة **serviceError** مع مشكلة **timeLimitExceeded** إلى DUA الطالب أو زبون LDAP أو DSA ويجرى إنهاء محول العملية. وبديلاً عن ذلك، قد يختار إجراء أن يناول هذا الحدث بطريقة مختلفة (مثل، خلال معالجة طلب **search**).

إذا استقبل DSA طلباً من DSA آخر مع حد وقت منفذ، يرسل **serviceError** مع مشكلة **timeLimitExceeded** دون أي معالجة أخرى للطلب.

وإذا كان لدى DSA طلبات (طلبات فرعية) معلقة، عندما ينقضي **timeLimit**، ولا تتاح نتائج، يعيد **serviceError** مع مشكلة **timeLimitExceeded** إلى الطالب.

وإذا كان لدى DSA طلبات فرعية معلقة، عندما ينقضي **timeLimit**، وتتاح نتائج، يعيد نتيجة إلى الطالب مع المحتويات التالية:

أ) جميع النتائج المجمعة حتى انقضاء **timeLimit**؛

ب) مكون **limitProblem** لـــ **partialOutcomeQualifier** تضبط معلمة-نتيجة على **timeLimitExceeded**.

ج) مكون **unexplored** لمعلمة-نتيجة **partialOutcomeQualifier** تحتوي على قيمة مرجع استمرارية لكل مجموعة من DSAs التي أرسلت طلبات فرعية ولكن النتيجة ليست شاملة في النتيجة إلى الطالب، بالإضافة إلى مراجع استمرارية لـــ DSAs حيث لم يحاول DSA هذا أن يرسل طلبات فرعية.

2.4.1.16 خسارة تصاحب

إذا تمت خسارة تصاحب الطالب، تتم خسارة جميع إمكانيات إعادة نتائج. ويمكن لـــ DSA اختيارياً أن يرسل لكل طلب (طلب فرعي) استفسار معلق طلب **chainedAbandon**، ما لم يكون التصاحب مع DSA قيد النظر قد تمت خسارته. ويجرى تجاهل جميع الإجابات على طلبات **chainedAbandon** وجميع الإجابات على الطلبات (الطلبات الفرعية). وفي حالة نتائج متصفحة لـــ DSP، ينبغي على DSA الموثق أن يلغي النتائج المتصفحة المعلقة بواسطة توليد طلب نتيجة متصفحة جديدة بواسطة اختيار **abandonQuery** لـــ **PagedResultsRequest**.

إذا تمت خسارة التصاحب لواحد من الطلبات الفرعية المتسلسلة المعلقة ولم تتم خسارة تصاحب الطالب، قد يحاول DSA اختيارياً، لعمليات الاستفسار فقط، أي مرجع بديل إلى DSA آخر قادر على معالجة طلب متسلسل (مثل، مرجع ظل DSA، وبعد خسارة التصاحب مع DSA الرئيسي). وإذا لم ينجح هذا، يعمل DSA كما يلي:

1) إذا ضبط **operationProgress.nameResolution** على **notStarted** أو **proceeding**، يعيد إما **serviceError** مع مشكلة **unavailable** إلى الطالب أو خطأ مرجع يحتوي على مرجع استمرارية مجموعة DSAs تتمكن من مواصلة العملية. وإذا استخدمت مراجع تابعة غير محددة خلال مرحلة NameResolution ولم يتم خسارة جميع التصاحبات قيد النظر، تحاول اختيارياً أن تقوم باستبانة اسم دون DSAs الذين خسروا التصاحبات. وإذا فشل هذا، يعاد إما **serviceError** مع مشكلة **unavailable** أو خطأ مرجع يحتوي على مجموعة NSSRs تامة.

إذا عرف DSA المستخدم لمعرفة محلية، من الممكن أن ينعكس في قيمة **MasterOrShadowAccessPoint** ملائمة، أن سلسلة مطلوبة لـــ DSA الذي خسر تصاحب، يختار أن يرسل **serviceError** مع مشكلة **unavailable** ومكون **notification** لـــ لنمط معطيات لـــ **CommonResults** يحتوي على:

- نعت تبليغ **dSAProblem** مع قيمة **id-pr-targetDsaUnavailable**؛
- نعت **distinguishedName** مع قيمة اسم مميز لـــ DSA.

2) إذا ضبط **operationProgress.nameResolution** على **completed** وكان الطلب عملية شيئ وحيد، يعاد **serviceError** مع مشكلة **unavailable** إلى الطالب.

3) إذا ضبط **operationProgress.nameResolution** على **completed** وكان الطلب عملية استفسار لمدخل متعدد، يضيف DSA مرجع استمرارية إلى **partialOutcomeQualifier.unexplored** لنتيجة العملية، مع **AccessPointInformation** محدداً مجموعة DSAs التي تتمكن من مواصلة العملية، بما في ذلك أي DSAs قد تمت خسارة تصاحباتهم.

3.4.1.16 ترك العملية

خلال معالجة عملية، يمكن استقبال طلب Abandon لهذه العملية. وفي هذه الحالة، خلال معالجة طلب **Abandon**، يطلب إجراء Abandon لترك العملية.

4.4.1.16 الحدود الإدارية

قد تكون هناك حدود يفرضها إداري DSA المحلي أو بواسطة تنفيذ DSA نفسه، مثل، كمية الوقت المستغرق في معالجة طلب، أو الحجم الأقصي للمعطيات التي تعاد وما إلى ذلك. وإذا تم تجاوز أي من هذه الحدود، يعيد DSA إما **serviceError** مع مشكلة **administrativeLimitExceeded** أو نتيجة جزئية (مأخوذة من مجموعة من نتائج مجمعة فعلاً) مع **limitProblem** مضبوطة على **administrativeLimitExceeded**.

تعاد معلومات اضافية في نعوت تبليغ **dSAProblem** كما يلي:

أ) إذا فرض الحد إداري، يأخذ نعت تبليغ **dSAProblem** قيمة **id-pr-administratorImposedLimit**؛

ملاحظة – لا يعني هذا أن التنفيذ مطلوب لتحقيق قدرات مشخصنة لإداري لفرض حدود إدارية.

ب) إذا تسبب في الحد قيد التنفيذ وتم إدراك المشكلة على أنها ذات طابع دائم، يأخذ نعت تبليغ **dSAProblem** قيمة **id-pr-permanentRestriction**؛

ج) إذا تسبب في الحد قيد التنفيذ وتم إدراك المشكلة على أنها ذات طابع مؤقت، مثل، ازدحام مؤقت، يأخذ نعت تبليغ **dSAProblem** قيمة **id-pr-temporaryRestriction**.

5.4.1.16 حد الحجم

يمكن تجاوز حد الحجم، كما ورد في في **CommonArguments**، عند أي نقطة من الزمن خلال معالجة عملية List أو Search. وفي هذه الحالة، تعاد نتيجة جزئية (مأخوذة من نتائج مجمعة فعلاً) إلى الطالب مع **limitProblem** مضبوطة على **sizeLimitExceeded**. وبالإضافة إلى ذلك، يمكن استخدام مكون **unexplored** لإعادة Continuation References لـــ DSAs غير النافذين.

إذا ضبطت عملية بحث وخيار مراقبة بحث **entryCount**، يضع DSA أفضل تقدير عن عدد المداخل التي يحتمل إعادتها إذا لم يكن هناك حد للحجم مع الأخذ في الاعتبار مراقبة النفاذ وليس الاختيارات التراتبية، ثم يعاد ذلك الرقم في مكون

entryCount لـــ **PartialOutcomeQualifier** باستخدام اختيار **bestEstimate** إذا لم يكن هناك DSAs **unaccessed**، وإلا يختار **lowEstimate**.

ثم يجرى إنهاء Operation Dispatcher.

2.16 إجراءات محول عمليات

يعرف الإجراء الذي يؤدى به **Operation Dispatcher** معالجة كل طلب مستقبل (عبر DAP أو LDAP أو DSP) بالخطوات التالية. ونتيجة لإبدال مستعار، يمكن أن يسمي هذا الإجراء نفسه (طلب محلي) وفي أي حالة تعاد إجابة محلية (بدلاً من إجابة DAP أو LDAP أو DSP).

1) قر بصلاحية جوانب عديدة لمتغير العملية (إجراء **Request Validation**). وإذا تمت مواجهة خطأ خلال إقرار الصلاحية، اعد هذا الخطأ محلياً أو عبر DSP/LDAP/DAP.

2) إذا كانت العملية المستقبلة عملية **Abandon**، استدعي إجراء **Abandon**، وأعد الإجابة بعد ذلك.

3) استبين اسم شيء مستهدف بواسطة تنفيذ إجراء **Find DSE** (الذي يتضمن الإجراءات الفرعية **Target Found** و**Target Not Found**). وإذا وجد المدخل المطلوب وهو مناسب (طبقاً لوضع مراقبة النفاذ ومتغيرات سلسلة وقرارات سياسة محلية)، استمر مع **Evaluation Phase** في الخطوة 6). وإذا تمت مواجهة خطأ خلال Name Resolution، يعاد. وإذا وجد أن المدخل غير مناسب، استمر مع الخطوة 4.

4) يستدعي إجراء **Name Resolution Continuation Reference** لمعالجة قائمة Continuation Reference كما تم تخزينها في **NRcontinuationList**. ولكي تعالج Continuation References هذه، يمكن إصدار طلبات متسلسلة إلى DSAs آخرين (إذا سمحت بها مراقبة خدمة وقرار سياسة محلية).

وفي حالة خطأ، يعاد هذا الخطأ مباشرة سواء محلياً أو عبر DSP/LDAP/DAP. وإذا ولد الطلب المتسلسل نتيجة، استمر مع الخطوة 5).

5) يستدعي إجراء **Result Merging** لدمج النتائج المحلية مع النتائج المتسلسلة المستقبلة. وإذا احتوت النتائج المتسلسلة على Continuation References مدمجة، يمكن استبانة هذه أولاً إذا كانت مراقبة خدمة والسياسة المحلية تسمح بها أو تتطلبها.

وقد يسبب هذا إصدار طلبات متسلسلة اضافية (قد تحتوي نتائجها المتسلسلة أيضاً على Continuation References مدمجة).

تعاد النتائج المدمجة إلى الطالب، وتتوقف معالجة الطلب.

وإذا تم أداء حماية النتائج، لا يؤدى دمج النتائج.

6) إذا كانت العملية هي عملية تعديل، استمر مع الخطوة 7.

إذا كانت العملية هي عملية استفسار مدخل وحيد، استمر مع الخطوة 8.

إذا كانت العملية هي عملية استفسار مدخل متعدد، استمر مع الخطوة 9.

7) عند تنفيذ إجراء تعديل، قد يتعين إنشاء Operational Binding، أو تعديله أو إنهاؤه، أو قد تحتاج الظلال إلى تحيينها كتتابع لأداء العملية. وسواء تم هذا تزامنياً أو لا تزامنياً مع أداء العملية الأصلية، يعتمد على عمليات التعديل المعنية (وعلى السياسة المحلية). وتعاد نتيجة أو خطأ محلي أو DSP/LDAP/DAP إلى الطالب.

8) تعاد نتيجة عملية استفسار مدخل وحيد مباشرة إلى الطالب باعتبارها نتيجة محلية أو لـــ DSP/LDAP/DAP.

9) إذا كانت العملية هي عملية استفسار لمدخل متعدد، تأكد من **nameResolutionPhase** للعملية. وإذا لم تكن **completed**، يستدعي إجراء **List(I)** أو **Search(I)**، وإلا استدعي إجراء **List(II)** أو **Search(II)** على التوالي.

10) يمكن إعادة ناتج نداء إجراء **List(II)** (نتيجة أو خطأ) وناتج نداء إجراء **List(I)** (في حالة أن الناتج هو خطأ) مباشرة إلى الطالب (باعتباره نتيجة محلية أو لـــ DSP/LDAP/DAP).

وإذا كان الإجراء المطلوب هو إجراء **List(I)**، قد تحتوي النتيجة Continuation References التي يتعين إبدالها (يعتمد على مراقبة الخدمة والسياسة المحلية). وقد ينتج عن هذا في عمليات List متسلسلة أن ترسل إلى DSAs المعنيين. ولدمج النتائج استمر مع الخطوة 5 مع نداء إلى إجراء **Result Merging**.

11) إذا كانت العملية هي عملية Search، تجرى استبانة أي Continuation References بواسطة إجراء **Search Continuation Reference** (إذا طلب أو سمح به). وقد يسبب هذا لطلبات Chained Search أن ترسل إلى DSAs المعنيين. ويطلب إجراء **Result Merging** [انظر الخطوة 5] لدمج نتائج البحث ومن الممكن إبدال Continuation References، إن وجدت.

3.16 نظرة شاملة على الإجراءات

يقدم هذا القسم نظرة شاملة على الوظيفية الأساسية للإجراءات التي يستخدمها **Operation Dispatcher** المعرفة في الأقسام من 17 إلى 22.

1.3.16 طلب إجراء إقرار صلاحية

يطلب هذا الاجراء، الوارد في القسم 17، لأداء التأكد من العروات والتأكد من الحد والتأكد من الأمن قبل أداء استبانة اسم محلي. ويوفر هذا الإجراء أيضاً بيئات بالتغيب لمعلمات **ChainingArgument** غير الموفرة من DAP أو LDAP في حالة أن الطلب جاء من DUA أو زبون LDAP. وفضلاً عن ذلك، يحدد هذا الإجراء أي طلب **abandon** ويبلغ هذا إلى **Operation Dispatcher**.

2.3.16 إجراء ترك

يحاول هذا الإجراء، الوارد في 5.20، أن يجد العملية التي تترك ويجرى إنهاؤها. وإذا كان هناك طلبات فرعية معلقة، يمكن إرسال عمليات Chained Abandon بعدها. ويعيد الإجراء إما نتيجة فارغة إلى الطالب أو دلالة خطأ (مثل، **abandonError** مع مشكلة **tooLate**).

3.3.16 إجراء Find DSE

يوائم هذا الإجراء، الوارد في 2.18 و3.18، مكونات اسم شيء مستهدف مقابل DSEs محتفظ بها محلياً لاستبانة اسم شيء مستهدف. وإذا تمت مواجهة DSE مستعار، يبدل المستعار (إذا سمح بذلك) ويعاد بدأ الإجراء لاستبانة الاسم الجديد.

وإذا لم يوجد الهدف، يتواصل الإجراء عند الإجراء الفرعي **Target Not Found**. وإذا وجد الهدف، يتواصل الإجراء عند الإجراء الفرعي **Target Found**.

ملاحظة – إن **Target Not Found** و**Target Found** هما استمرارية لإجراء **Find DSE**.

قد ينتج الإجراء أخطاءً مختلفة، وفي هذه الحالة، يعاد البروتوكول المصاحب إلى الطالب ويجرى إنهاء **Operation Dispatcher**.

1.3.3.16 الإجراء الفرعي Target Not Found

يؤدى هذا الإجراء، الوارد في 2.3.18، تقييماً لـــ DSEs الوسيطة المحددة مواقعها ويخلق مجموعة من Continuation References في **NRcontinuationList**، على أساس مجموعة مراجع معرفة تم اكتشافها خلال إجراء **Find DSE**. ثم تعالج أيضاً مجموعة المراجع هذه في إجراء **Name Resolution Continuation Reference**.

قد ينتج الإجراء أخطاء مختلفة، وفي هذه الحالة، يعاد الخطأ المصاحب إلى الطالب ويجرى إنهاء **Operation Dispatcher**.

2.3.3.16 الإجراء الفرعي Target Found

يتأكد هذا الإجراء، المعرف في 3.3.18، إذا كان DSE الذي وجد مناسباً للعملية المطلوبة، أي، في حالة معلومات مظللة. وقد يشمل هذا التأكد من ملائمة الشجرة الفرعية بكاملها للمعلومات المظللة تحت الشيء المستهدف في حالة عملية شيء متعدد (مثل، بحث شجرة فرعية).

وإذا كان المدخل المحدد موقعه مناسباً، يتم تنفيذ إجراء تقييم العملية الملائم. وإلا، يجرى إنشاء **ContinuationReference** مشيراً إلى مورد (أو رئيسي) المعلومات في **NRcontinuationList** وينفذ إجراء **Name Resolution Continuation Reference**.

4.3.16 إجراء استفسار مدخل وحيد

ينفذ هذا الإجراء، الوارد في 2.19، للتنفيذ الفعلي لتلك العمليات التي تؤثر فقط على مدخل وحيد، أي، عمليتي Read وCompare. وبعد الاكتمال، تعاد إجابة (نتيجة أو خطأ) يخلقها الإجراء إلى DSA/DUA/زبون LDAP الطالب.

5.3.16 إجراءات تعديل

تنفذ هذه الإجراءات، الواردة في 1.19، لمعالجة عمليات تعديل،أي، Add Entry وRemve Entry، Modify DN وModify Entry. ويتم هذا بواسطة تنفيذ إجراء فرعي محدد معرف لكل من هذه العمليات. وخلال (أو بعد) الإجراءات الفرعية هذه، يمكن إصدار طلبات DOP وDISP إلى DSAs آخرين. وبعد الاكتمال بنجاح، تعاد نتيجة (خلقتها إجراءات فرعية) إلى DSA/DUA/زبون LDAP الطالب.

6.3.16 إجراءات استفسار مدخل متعدد

تنفذ هذه الإجراءات، الواردة في 3.19، لمعالجة عمليات تؤثر على مداخل متعددة قد تحدد أو لا تحدد مواقعها في نفس DSA. ويتم هذا بواسطة تنفيذ إجراءات فرعية محددة معرفة لكل من عمليات Search وList لانجاز تجزئة طلب. وتخلق هذه الإجراءات نتيجة محلية لتقييم العملية واختيارياً مجموعة من مراجع استمرارية في **SRcontinuationList**. وإذا كانت **SRcontinuationList** فارغة عند طرف هذا الإجراء، تعاد النتيجة التي خلقت مباشرة إلى DSA/DUA/زبون LDAP الطالب. وإذا كانت عملية Search، وإذا كانت النتيجة فارغة وإذا ضبط متغير **emptyHierarchySelect**، تعاد في مكون **notification** لـــ **PartialOutcomeQualifier**:

- نعت تبليغ **searchServiceProblem** مع قيمة **id-pr-emptyHierarchySelection**.

وإذا كانت **SRcontinuationList** غير فارغة، تعالج مراجع الاستمرارية هذه بواسطة تنفيذ **List** أو **Search Continuation Reference**، طبقاً لنمط العملية.

7.3.16 إجراء Name Resolution Continuation Reference

يعالج هذا الإجراء، الوارد في 1.4.20، مراجع الاستمرارية في **NRcontinuationList** الذي خلق خلال مرحلة Name Resolution. وتستخدم مراجع الاستمرارية هذه إما لاصدار طلبات فرعية متسلسلة أو تعاد في خطأ مرجع. وفي حالة السلسلة، تعاد النتائج أو الأخطاء المعادة من الطلب المتسلسل لمزيد من المعالجة بواسطة إجراء **Result Merging**.

8.3.16 إجراءات List وSearch Continuation Reference

تعالج الإجراءات هذه، الواردة في 2.4.20 و3.4.20، مراجع الاستمرارية في **SRcontinuationList** االتي خلقتها إجراءات استفسار مدخل متعدد سواء لاستبانتها بواسطة اصدار طلبات فرعية متسلسلة أو بواسطة خلق مرجع (مراجع) استمرارية في **partialOutcomeQualifier.unexplored**. وعند استقبال نتائج أو أخطاء جميع الطلبات الفرعية المعلقة، تعاد لمزيد من المعالجة بواسطة إجراء **Result Merging**.

9.3.16 إجراء Result Merging

يفحص هذا الإجراء، الوارد في القسم 21، إما النتيجة من طلب متسلسل أو تركيبات نتائج عملية محلية مع نتائج مستقبلة من طلبات فرعية متسلسلة. وإذا أعاد طلب فرعي خطأ، يحدد هذا الإجراء كيفية مناولة هذا الخطأ.

وإذا كانت هناك أي مراجع استمرارية تركت في النتيجة، تبدل (إذا سمحت السياسة المحلية بذلك وتتطلبها مراقبة خدمة) بواسطة إجراءات **Name Resolution** أو **List** أو **Search Continuation Reference**، بناء على ذلك. وتسحب النسخ المطابقة من النتيجة إذا لم تكن موقعة.

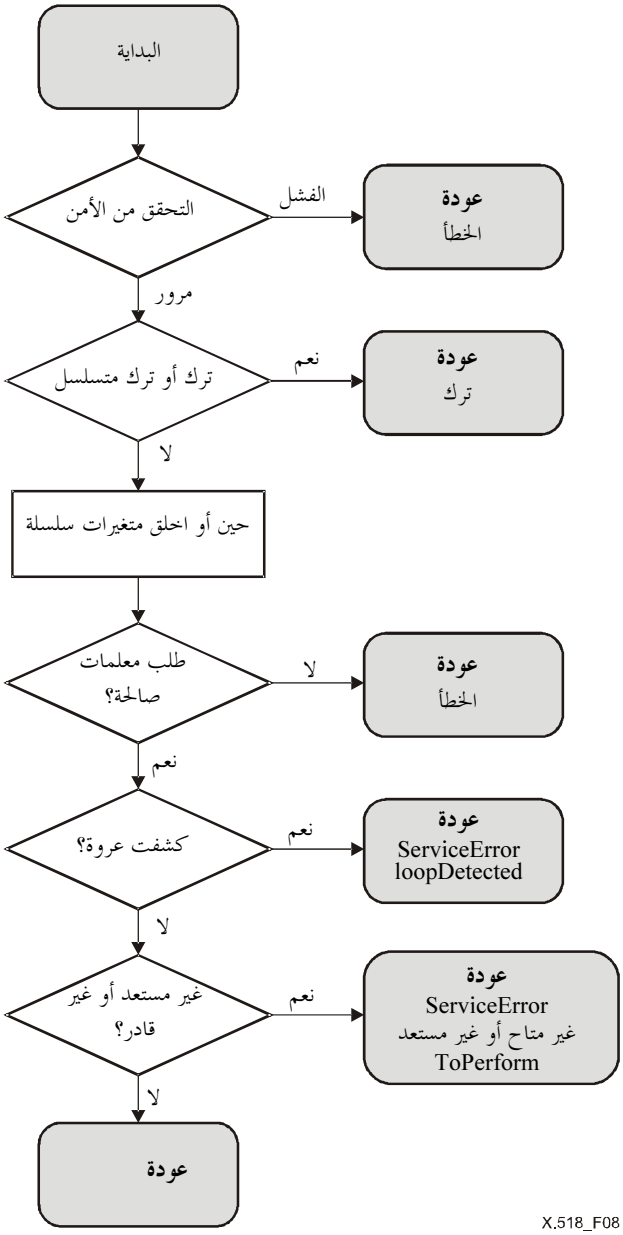
تعاد النتيجة المدمجة (مع جميع النتائج المدمجة ومراجع الاستمرارية غير المستبانة) إلى DUA//زبون DSA/LDAP الطالب.

وإذا تم أداء حماية على النتائج، لا يجرى أداء دمج النتائج.

17 إجراء اقرار صلاحية طلب

1.17 مقدمة

إن إجراء **Request Validation** هي نقطة المدخل لـــ Operation Dispatcher لمدخلات من DUA//زبون DSAs/LDAP ، لإعداد هذه المدخلات لمعالجة Name Resolution. ووظيفة هذا الإجراء هي كشف عمليات ترك، ولأداء التحقق من الأمن، وتسوية الدخل المستقبل من DUAs أو زبائن LDAP حتى يمكن معالجته بنفس الطريقة باعتباره دخلاً من DSAs، وللتحقق من متغيرات الطلب لقواعد تركيب وعلم دلالات صالح، ولأداء كشف عروة، ولأداء عمليات تحقق متنوعة أخرى. ويرد في الشكل 8 تدفق **Request Validation**.



الشكل 8 – إجراء اقرار صلاحية طلب

2.17 معلمات إجراء

1.2.17 متغيرات

إن متغير دخل لـــ **Request Validation**. يتألف من **ChainingArguments** (باستثناء في حالة عمليات **chainedAbandon**)، إذا استقبل الطلب من DSA، وأصدر مرسل الطلب المتغير.

2.2.17 نتائج

تتألف نتيجة خرج **Request Validation** من خمس إمكانيات:

أ) إذا فشل التحقق من الأمن، يعاد خطأ إلى الطالب.

ب) إذا كان الدخل عملية **abandon** أو **chainedAbandon**، يكون الخرج هو متغير العملية.

ج) إذا كانت متغيرات الطلب غير صالحة، يعاد خطأ إلى الطالب. واعتماداً على السياسة المحلية، قد يختار DSA ما إذا كان يعيد **serviceError** أو **securityError**.

د) إذا كشفت عروة، يعاد **serviceError** مع **loopDetected** إلى الطالب.

هـ) على أساس مشاكل المورد أو اعتبارات السياسة، إذا لم يكن DSA قادراً أو غير مستعد لأداء عملية، يعاد **serviceError** (مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform**) إلى الطالب. وإذا تعلق الأمر، يمكن إعادة **serviceError** مع مشكلة **dataSourceUnavailable**.

و) وفي جميع الحالات، يكون الدخل الذي اقرت صلاحيته، المحول بواسطة **ChainingArguments** اضافية إذا استقبلت من DUA أو زبون LDAP أو تحيين **ChainingArguments.traceInformation** إذا استقبلت من DSA، هو خرج الإجراء وبالتالي الدخل إلى إجراء **Name Resolution**.

3.17 تعريف إجراء

يجرى أداء التحقق من الأمن الوارد في 2.3.17. ويمكن أن ينتج عن هذا إعادة خطأ وإنهاء Operation Dispatcher.

إذا كان الدخل عملية **abandon** أو **chainedAbandon**، تؤدى الخطوات في 1.3.17 فقط فيما بعد، وإلا تؤدى الخطوات في 3.3.17–5.3.17. ويصف القسم الفرعي 5.3.17 إجراء كشف عروة الذي يمكن أن ينتج عنه إعادة خطأ وإنهاء Operation Dispatcher.

ثم يؤدى التحقق في 6.3.17. ويمكن أن ينتج عنه إعادة خطأ وإنهاء Operation Dispatcher.

إذا لم ينتج عن التحقق في 2.3.17–6.3.17 إنهاء Operation Dispatcher، يؤدى التحقق في 7.3.17 ويجرى إنهاء الإجراء مع نقل خرجه إلى إجراء **Name Resolution**.

1.3.17 معالجة ترك

يمرر متغير **abandon** أو **chainedAbandon** إلى إجراء **Abandon**، (انظر 5.20) لمعالجة طلب ترك.

2.3.17 التحقق من الأمن

إذا وقع أو جفر متغير العملية أو وقع وجفر، يمكن التحقق من التوقيع. وإذا كان التوقيع غير صالح أو فشل التجفير أو كان غائباً في حالة وجوب أن يكون معيِّناً، قد يعاد خطأ إلى الطالب. وبديلاً عن ذلك، يمكن أن يؤدى DSA أي عمل آخر معرف محلياً.

3.3.17 إعداد دخل

1.3.3.17 طلب DUA أو زبون LDAP

إذا استقبلت العملية من قبل DUA أو زبون LDAP، تنشأ قيمة **ChainingArguments** كما يلي:

أ) يضبط **ChainingArguments.originator** كما ورد في 3.10.

ب) يضبط **ChainingArguments.operationProgress** على قيمة **CommonArguments.operationProgress**.

ج) يضبط **ChainingArguments.traceInformation** على تتابع يحتوي على قيمة **TraceItem** وحيد. وتبني هذه القيمة كما يلي. يضبط **TraceItem.dsa** على اسم DSA المنفذ لـــ **Request Validation**. وتحذف **TraceItem.targetObject**. ويضبط **TraceItem.operationProgress** على القيمة الواصلة.

د) إذا حددت مراقبة الخدمة للعملية حد الوقت (الوقت المنقضي المتاح بالثواني لاكتمال العملية)، يضبط **ChainingArguments.timeLimit** على وقت (UTC) الذي بمقتضاه تكتمل العملية لتلبية حد الوقت الذي حدده المستعمل.

ه) يضبط **ChainingArguments.AuthenticationLevel** و**ChainingArguments.UniqueIdentifier** طبقاً لسياسة الأمن المحلية.

و) ينسخ ChainingArguments.nameResolveOnMaster من CommonArguments.nameResolveOnMaster.

ز) ينسخ **ChainingArguments.exclusions** و**ChainingArguments.entryOnly** و**ChainingArguments.referenceType** من **CommonArguments.entryOnly** و**CommonArguments.referenceType** إذا كان لهم وجود وإلا، يحذفون.

ح) إذا ضبط خيار **manageDSAIT** في **ServiceControls**، فإن:

- مكون **nameResolutionPhase** لـــ **operationProgress** يضبط على **completed**؛
- مكون **nextRDNToBeResolved** لـــ **operationProgress** يحذف ؛
- **referenceType** يأخذ قيمة **self**؛
- **entryOnly** يأخذ قيمة **FALSE**؛
- **nameResolveOnMaster** يأخذ قيمة **FALSE**؛
- يضبط خيار **ChainingProhibited** في **ServiceControls** ؛
- العناصر الاختيارية المتبقية لـــ **ChainingArguments** تحذف، مع قيمها بالتغيب التي تفترض حيثما تحدد.

ط) إذا لم يضبط خيار **manageDSAIT** في **ServiceControls**، تحذف العناصر الاختيارية المتبقية لـــ **ChainingArguments**، مع قيمها بالتغيب التي تفترض حيثما تحدد.

ي) يستخدم **ChainingArguments.SecurityParameters.ProtectionRequest** ليدل على مستوى الحماية (يوقع أو يجفر أو يوقع ويجفر) الذي يطبق على النتائج.

2.3.3.17 طلب LDAP

إذا استقبلت العملية من زبون LDAP، تنشأ قيمة **ChainingArguments** حسب القسم 1.3.3.17، مع استثناء أن يضبط **ChainingArguments.operationProgress** على **nameResolutionPhase notStarted**، وتحذف قيم **ChainingArguments.exclusions** و**ChainingArguments.entryOnly** و**ChainingArguments.referenceType**.

3.3.3.17 طلب DSA

إذا استقبلت العملية من DSA، يحين **ChainingArguments.traceInformation** بواسطة تذييل قيمة في طرف تتابع **TraceItem**. وتبني هذه القيمة كما يلي:

أ) يضبط **TraceItem.dsa** على اسم DSA المنفذ لـــ **Request Validation**.

ب) يضبط **TraceItem.targetObject** على قيمة **ChainingArguments.targetObject** ما لم يكن **object** (أو **baseObject** في حالة عملية Search) لمتغير طلب مماثل لـــ **ChainingArguments.targetObject**، وعلى أي حال يحذف **TraceItem.targetObject**.

ج) يضبط **TraceItem.operationProgress** على قيمة **ChainingArguments.operationProgress**.

إذا استقبلت العملية من DSA، وإذا احتوت **ChainingArguments.streamedResults** على قيمة أكبر من أو مساوية لواحد، وإذاً وفقط إذا فهم DSA النتائج المتدفقة وكان على استعداد لقبول النتائج المتدفقة لهذه العملية، تزداد قيمة **ChainingArguments.streamedResults** بواحد.

4.3.17 توكيد الصلاحية

يجرى التحقق من العملية من أجل قواعد تركيب وعلم دلالات صالح لمتغيراتها طبقاً للقواعد الواردة في الأقسام التي تعرف كل عملية (مثلاً، ينبغي التحقق من أن **nextRDNToBeResolved** لا يوفر عدداً لا يتجاوز عدد RDNs في **targetObject**). وإذا اكتشف أن الطلب يحتوى على متغيرات غير صالحة، يجرى إنهاء العملية ويعاد خطأ إلى المستعمل، يعتمد على نوع عدم الصلاحية المكتشف.

5.3.17 اكتشاف عروة

إذا كانت أي قيمتين لـــ **TraceItem** لـــ **ChainingArguments.traceInformation** (كما أعد في 3.3.17) متماثلتين، تعاد معالجة العملية إلى الحالة السابقة، أي، قد تم اكتشاف عروة. وفي هذه الحالة، يعاد **serviceError** (مع مشكلة **loopDetected**) إلى الطالب ويجرى إنهاء **Operation Dispatcher**.

6.3.17 غير قادر أو غير مستعد للأداء

يمكن أن يقيم **Request Validation** ويحدد أن العملية لا يمكن أداؤها. وقد يحدد أيضاً، على أساس اعتبارات السياسة، أن العملية ينبغي ألا تؤدى. وفي هذه الحالات، يمكن إعادة **serviceError** (مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform**) إلى الطالب ويجرى إنهاء Operation Dispatcher.

إذا استطاع DSA بواسطة وسائل محلية أن يحدد أن المشكلة متعلقة بعدم توافر موارد محلية لـــ DIB، يرسل **serviceError** مع مشكلة **unavailable** ويحتوى مكون **notification** لنمط معطيات **CommonResults** على:

- نعت تبليغ **dSAProblem** مع قيمة **id-pr-dataSourceUnavailable**؛
- نعت **distinguishedName** له قيمة اسم مميز لـــ DSA.

7.3.17 معالجة خرج

في المرحلة النهائية لـــ **Request Validation**، يعاد دخل تم إقرار صلاحيته، محول بواسطة **ChainingArguments** إذا استقبل من DUA أو زبون LDAP، أو تحيين **ChainingArguments.traceInformation** إذا استقبل من DSA، ويستخدم كدخل إلى إجراء **Name Resolution**.

18 إجراء Name Resolution

1.18 مقدمة

يصف هذا القسم إجراء **Name Resolution**، ومتغيراته ونتائجه وشروط خطأ ممكن. وكما يرد في الشكل 6 (**Operation Dispatcher**)، يتألف **Name Resolution** من متغيرين.

- إجراء **Find DSE**؛

– إجراء **Name Resolution Continuation Reference**.

يوصف إجراء **Find DSE** في ثلاثة مخططات انسيابية، أي **Find DSE** و**Target Found** و**Target Not Found**. ويوائم إجراء **Find DSE** اسم مدخل مستهدف لـــ DSEs مخزنة محلياً، كل مكون على حدة. وإذا وجد المدخل المستهدف محلياً، يستمر **Find DSE** مع الإجراء الفرعي **Target Found**، الذي يستدعي إجراء **Check Suitability** للتحقق من ملاءمة **Find DSE** للتقييم. وإذا لم يوجد المدخل المستهدف محلياً، فإن **Find DSE** يستمر مع الإجراء الفرعي **Target Not Found** ويعد إضافة Continuation Reference(s) إلى **NRcontinuationList** لإجراء **Name Resolution Continuation Reference** لتحويله.

الملاحظة 1 – يؤدى **Name Resolution** مواءمة اسم مقابل قيم متميزة متعددة متفاضلة بواسطة سياق، كما ورد في 4.9 من ITU-T X.501/ISO/IEC 9594-2، عندما يقرر المواءمة.

الملاحظة 2 – قد يفشل **Name Resolution** إذا احتفظ DSA علوي قبل الطبعة الثالثة بمرجع تابع لمدخل محتفظ به في DSA لطبعة لاحقة ويشمل RDN لذلك المدخل سياقات. ويفشل **Name Resolution** مقابل نسخة ظل لمدخل عندما يستخدم اسم بديل كاسم مستهدف ومدخل الظل محتفظ به في DSA لطبعة أولى أو ثانية.

2.18 معلمات إجراء Find DSE

1.2.18 متغيرات

يستخدم الإجراء المتغيرات التالية:

a) **ChainingArguments.traceInformation**;
b) **ChainingArguments.aliasDereferenced**;
c) **ChainingArguments.aliasedRDNs**;
d) **ChainingArguments.excludeShadows**;
e) **ChainingArguments.nameResolveOnMaster**;
f) **ChainingArguments.operationProgress** (**nameResolutionPhase**, **nextRDNToBeResolved**);
g) **ChainingArguments.referenceType**;
h) **ChainingArguments.targetObject**;
i) **ChainingArguments.relatedEntry**;
j) **ChainingArguments.streamedResults**;
k) the operation type;
l) the operation argument.

ملاحظة – عندما لا توجد قيم فعلية، تستخدم قيم بالتغيب أو ضمنية، كما ورد في 3.10.

2.2.18 نتائج

توجد حالتان لخرج ناجح من **Find DSE** (مشار إليهما بواسطة **entry suitable** أو **entry unsuitable**):

تعيد الحالة الناجحة الأولى (من الإجراء الفرعي **Target Not Found**) Continuation Reference(s) في **NRcontinuationList** التي تمرر إلى إجراء **Name Resolution Continuation Reference** لاستمرار مرحلة Name Resolution.

تعيد الحالة الناجحة الثانية (من الإجراء الفرعي **Target Found**) (مرجع إلى) DSE، الذي يمرر إلى واحد من إجراءات Evaluation.

3.2.18 أخطاء

يمكن إعادة الأخطاء التالية:

a) **serviceError**: **unableToProceed**, **invalidReference**, **unavailableCriticalExtension**, **requestedServiceNotAvailable**;

b) **nameError**: **noSuchObject**, **aliasDereferencingProblem**, **contextProblem**.

4.2.18 متغيرات اجمالية

يستخدم الإجراء المتغيرات الاجمالية التالية:

- قائمة **NRcontinuationList** لتخزين Continuation Reference(s) المطلوبة لاستمرار استبانة اسم في إجراء **Name Resolution Continuation Reference**.
- **StreamedResultsOK** لتخزين تحديد ما إذا كان DSA هذا قد يسلسل نتائج متدفقة في الاستجابة لهذه العملية.

5.2.18 متغيرات محلية ومتقاسمة

يستخدم الإجراء المتغيرات المحلية التالية:

أ) **i** دليل يستخدم لتعريف مكون اسم مستهدف يجرى العمل عليه.

ب) **m** طول اسم شيء مستهدف يستخدم في استبانة اسم. وللعمليات التي تستبين اسم لمدخل رئيسي، أي Add Entry، تضبط **m** (عدد RDNs في الشيء المستهدف) على واحد. ولجميع العمليات الأخرى، تضبط **m** على عدد RDNs في الشيء المستهدف.

ج) **lastEntryFound** دليل، بحيث يكون DSE (lastEntryFound) هو آخر DSE متوائم من نمط **entry**.

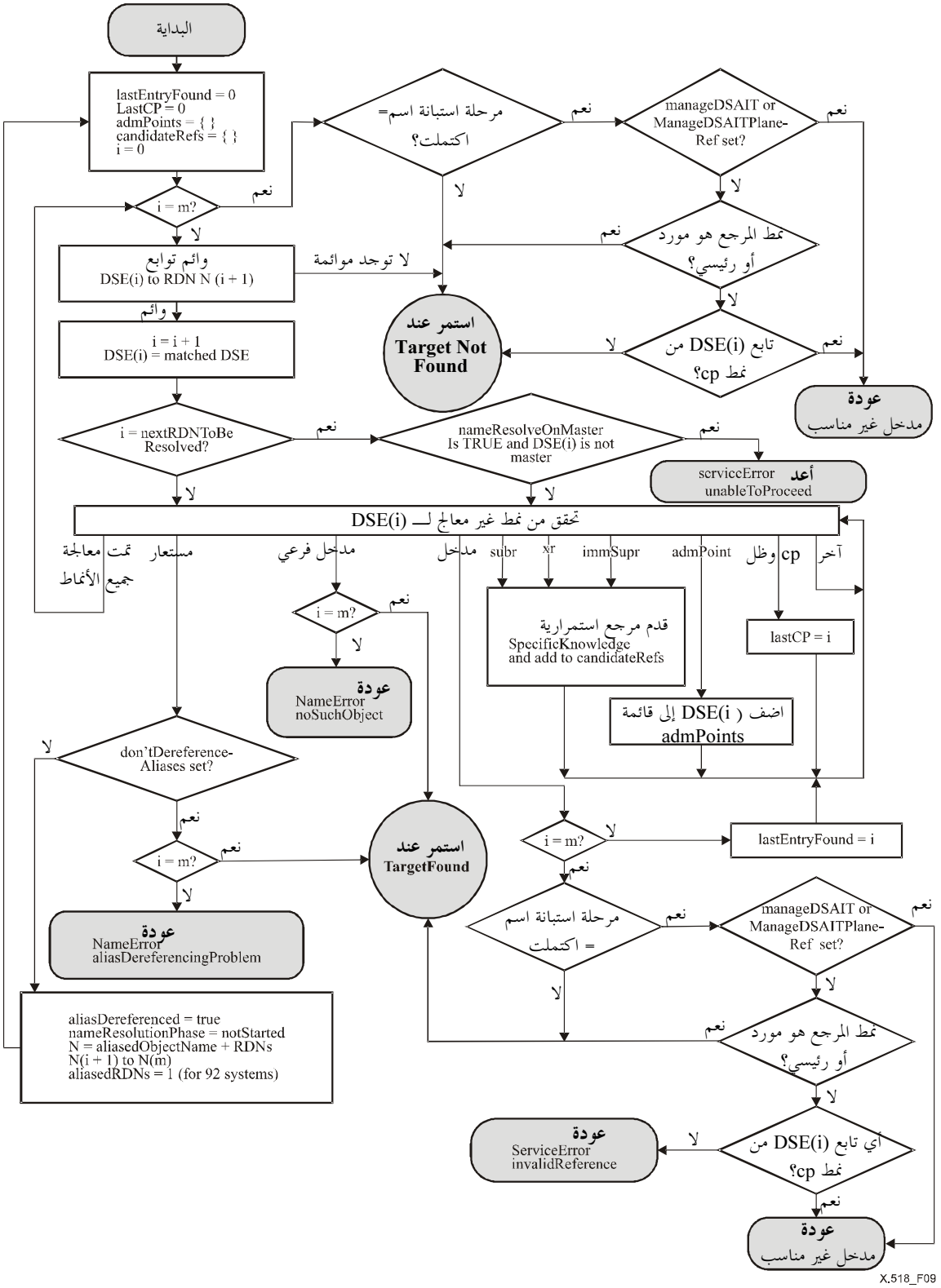
د) **lastCP** دليل، بحيث يكون DSE (lastCP) هو آخرسابقة سياق مظلل تمت مواجهتها.

هـ) **candidateRefs** مجموعة من مراجع استمرارية.

يستخدم أيضاً متغير متقاسم **admPoints** (المعرف في Operation Dispatche). وللتسهيل، يعرف مكون **i** لاسم شيء مستهدف على أنه **N(i)**.

3.18 إجراءات

ملاحظة – توجد بعض النصوص في المخطط الانسيابي المتعلق فقط بعمليات محددة. ولا يظهر هذا في المخططات الانسيابية، ولكن يوصف في النص المصاحب.



الشكل 9 – إجراء Find DSE

يحدد اسم شيء مستهدف كما يلي:

أ) إذا كان **targetObject** محيناً في **ChainingArguments**، تستخدم قيمة ذلك المكون.

ب) إذا كان **relatedEntry**، وليس **targetObject**، محيناً في **ChainingArguments**، يستخدم مكون **baseObject** لـــ **JoinArgument** المعرف بواسطة **relatedEntry**.

الملاحظة 1 – يتعلق هذا فقط بطلب **search** محمي.

ج) وإذا لم يكن **relatedEntry** ولا **targetObject** محيناً في **ChainingArguments**، يستخدم مكون **base (baseObject)** لمتغير العملية.

يحاول هذا الاجراء استبانة اسم شيء مستهدف محلياً:

1) تدميث المتغيرين المحليين **lastEntryFound** و**lastCP** على 0؛ **admPoints** و**candidateRefs** على مجموعة فارغة، وتدميث i على 0.

2) قارن **i** و**m**. وإذا كانا غير متماثلان، استمر عند الخطوة 5.

3) وإذا كانا متماثلان، تحقق من أن **nameResolutionPhase** هو **completed**. وإذا لم يكن **completed**، استمر عند الإجراء الفرعي **Target Not Found**.

إذا كان **nameResolutionPhase** هو **completed** والتمديد الحرج لـــ **manageDSAIT** مضبوطا، يعاد مع **entry suitable**.

4) إذا كان **nameResolutionPhase** هو **completed**، تحقق إذا كان أي تابع مباشر لـــ DSE(i) هي سابقة سياق (من نمط **cp**).

- إذا كان تابع مباشر واحد أو أكثر لـــ DSE(s) هو من نمط **cp**، يعاد مع **entry suitable**.

الملاحظة 2 – هذه الحالة للطلبات الفرعية لـــ **List (II)** و**Search (II)**.

- إذا لم تكن التوابع المباشرة لـــ DSE(i) هي من نمط **cp**، استمر عند الإجراء الفرعي **Target Not Found**.

5) حاول أن تجد مواءمة لمكون (i+1)-th لاسم شيء مستهدف مع اسم تابع لآخر DSE متوائم. وفي حالة i= 0، حاول أن توائم تابع مباشر واحد لـــ DSEs مع جذر DSE. وإذا لم توجد أي مواءمة، استمر عند الإجراء الفرعي **Target Not Found**. وإذا وجدت مواءمة واحدة، زدْ i وخزن DSE المتوائم مثل عنصر i-th في عائل DSEs الموجودة.

الملاحظة 3 – تشمل مواءمة اسم مناولة قيم متميزة متعددة متفاضلة بواسطة سياق، عندما تعرف، كما ورد في ITU-T X.501/ISO/IEC 9594-2 4.9.

وإذا وجدت أكثر من مواءمة واحدة، يعاد **nameError** مع مشكلة **contextProblem**.

الملاحظة 4 – فمثلاً، يمكن أن تكون هذه الحالة عندما يحتوى **AttributeTypeAndDistinguishedValue** في اسم مستهدف قيم نعت مميز متعدد متفاضل بواسطة سياقات ومختلفة عن القيم التي توائم قيم في أسماء مستهدفة مختلفة.

6) إذا كانت i مساوية لـــ **nextRDNToBeResolved**، تحقق إذا كان الشرطان قد تم تلبيتهما:

- **ChainingArgument.nameResolveOnMaster** هو **TRUE**؛
- DSE(i) ليس مدخلاً رئيسياً.

إذا تمت تلبية الشرطين، يعاد **serviceError** مع مشكلة **unableToProceed**.

الملاحظة 5 – يدل هذا على أن استخدام **nameResolveOnMaster** هو لتجنب مسيرات متعددة لنفس شيء مستهدف.

7) تحقق من جميع بتات نمط DSE لـــ DSE(i). ولكل بتة نمط، قد تطلب بعض المعالجة. ويرد أدناه العمل الذي يتخذ لكل نمط وجد:

- إذا ضبطت كل من بتات **cp** و**shadow**، تذكر أن الدليل **i** في **lastCP**.

- إذا ضبطت بتة **admPoint**، تحقق من النعت التشغيلي **administrativeRole**. وإذا كان هذا بداية منطقة إدارية مستقلة ذاتياً، عليك افراغ قائمة **admPoints**. وإذا كان هذا بداية منطقة أو أكثر إدارية محددة، تحقق من قائمة **admPoints** واسحب أي نقاط موجودة لم تعد ذات علاقة (أي، تم إحلال أدوارها بواسطة نقطة إدارية جديدة). وخزن DSE(i) في القائمة.

- إذا ضبطت بتة من بتات **subr** أو **xr** أو **immSupr** أو **ditBridge**، ولد مرجع استمرارية باستخدام نعت **specificKnowledge** مع **operationProgress.nameResolutionPhase** مضبوط على **proceeding**، و**nextRDNToBeResolved** مضبوط على **i**، ويبني **targetObject** من المكونات المستبانة باستخدام RDNs (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات الباقية غير المستبانة، وتضبط **accessPoints** و**referenceType** حسب الاقتضاء. واضف مرجع استمرارية إلى قائمة مراجع استمرارية في **candidateRefs**.

- إذا ضبطت بتة **entry**، اختبر إذا كانت **i** مساوية لـــ **m** (وبالتالي يجرى مواءمة اسم شيء مستهدف بالكامل). وإذا لم تكن **i** مساوية لـــ **m**، تذكر المدخل الذي وجد بواسطة ضبط **lastEntryFound** على **i** واستمر في معالجة بتات نمط DSE(i). وإذا كان كل من **i** و**m** متساويان، استمر عند الخطوة 8.

- إذا ضبطت بتة **subentry**، اختبر إذا كانت **i** مساوية لـــ **m** (وبالتالي يجرى مواءمة اسم شيء مستهدف بالكامل). وإذا كانا متساويان، استمر عند إجراء **Target Found**؛ وإذا لم يكونا متساويان، يعاد **nameError** مع مشكلة **noSuchObject**.

- إذا ضبطت بتة **alias**، اختبر إذا كان **dontDereferenceAliases** مضبوطاً.

  إذا لم يكن **dontDereferenceAliases** مضبوطاً، يمكن إبدال مستعار. ولهذا، اضبط **chainingArguments.aliasDereferenced** على **TRUE** و**nameResolutionPhase** على **notStarted**، واسم الشيء المستهدف على **aliasedEntryName** كما تم توريده في مدخل مستعار متسلسل مع المكونات المتبقية غير المتوائمة لاسم شيء مستهدف سابق (أي، متسلسل مع مكون th-(**i+1**) لاسم شيء مستهدف سابق). ولا تضبط الطبعة الثانية واللاحقة لـــ DSAs **aliasedRDNs** (بينما الطبعة الأولى لـــ DSAs تضبط **aliasedRDNs** على عدد RDNs في **aliasedEntryName**). ابدأ Name Resolution مرة ثانية بواسطة الاستمرار عند الخطوة 1.

  إذا ضبط **dontDereferenceAliases**، لا يمكن إبدال مستعار. وتحقق إذا كان اسم شيء مستهدف قد تمت معالجته تماماً بواسطة مقارنة **i** و**m** من أجل المساواة. وإذا كانا متساويان، (والاسم بالتالي متوائم بالكامل)، استمر عند الإجراء الفرعي **Target Found**. وإذا لم يكونا متساويان (والاسم بالتالي غير متوائم بالكامل)، يعاد **nameError** مع مشكلة **aliasDereferencingProblem**.

- بالنسبة لجميع أنماط DSE الأخرى الممكنة، ليست هناك حاجة لإجراء أي عمل. وداخلياً، لاحظ أن نمط DSE كما عولج ويستمر في معالجة بتات نمط DSE التي لم تعالج لـــ DSE(i).

- إذا تمت معالجة جميع بتات نمط DSE(i)، استمر عند الخطوة 2).

8) تحقق إذا كان **nameResolutionPhase** هو **completed**. وإذا لم يكن، استمر عند الإجراء الفرعي **Target Found**.

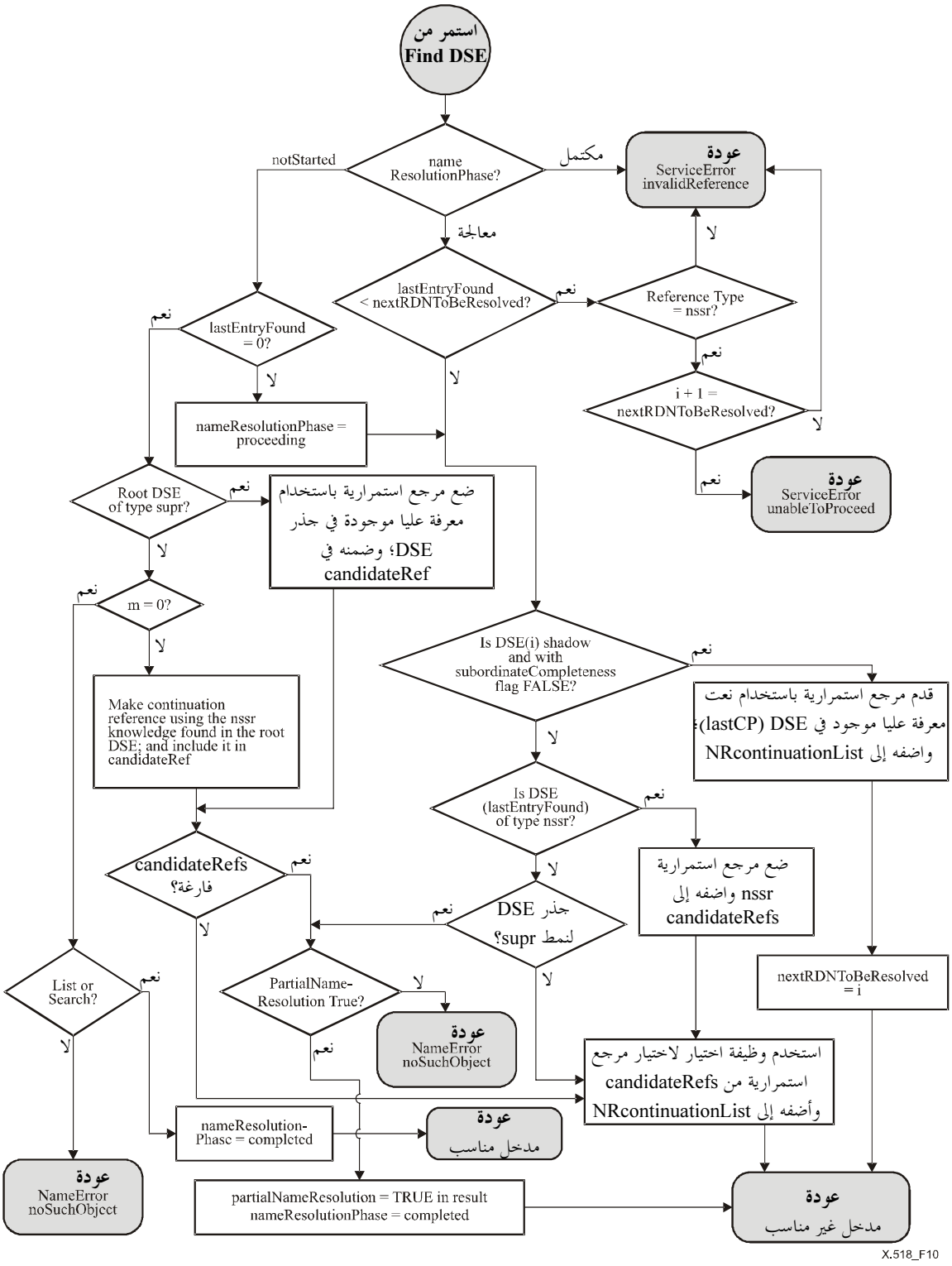
9) إذا كان **nameResolutionPhase** هو **completed** والتمديد الحرج لـــ **manageDSAIT** مضبوطاً، يعاد مع **entry suitable**.

10) وإلا، تحقق إذا كان أي تابع مباشر لـــ DSEs لـــ DSE(i) هي سابقة سياق (وبالتالي من نمط CP). وإذا كان هناك (واحد أو أكثر) يعاد **entry suitable**. وإذا لم يكن أي من مداخل تابع مباشر هو من نمط سابقة سياق، يعاد **serviceError** مع مشكلة **invalidReference**.

الملاحظة 6 – هذه الحالة للطلبات الفرعية لـــ **List (II)** و **Search (II)**.

2.3.18 الإجراء الفرعي Target Not Found

انظر الشكل 10.



الشكل 10 – الإجراء الفرعي Target Not Found

يستدعي الإجراء الفرعي هذا عندما لا يوجد اسم شيء مستهدف في DSA محلي. ويحدد الإجراء الفرعي هذا أفضل نمط مرجع معرفة ليستخدم لاستمرار استبانة اسم، ما لم يكتشف خطأ وفي هذه حالة يعاد الخطأ.

1) عند الاستمرار من إجراء **Find DSE**، ميز بين المراحل الثلاث الممكنة لمرحلة Name Resolution.

- إذا كان **nameResolutionPhase** هو **notStarted**، استمر عند الخطوة 2).
- إذا كان **nameResolutionPhase** هو **proceeding**، استمر عند الخطوة 8).
- إذا كان **nameResolutionPhase** هو **completed**، استمر عند الخطوة 12).

2) إذا وجد أي مدخل (**lastEntryFound** لا يساوي 0) اضبط **nameResolutionPhase** على **proceeding** واستمر عند الخطوة 9.

3) إذا لم يوجد أي مدخل (**lastEntryFound=0**)، تحقق إذا كان DSA هو First Level DSA.

وإذا كان First Level DSA، لا يحتوى جذر DSE Superior Reference وبالتالي ليس نمط **supr**. وفي هذه الحالة، استمر عند الخطوة 4.

وإذا كان DSA ليس First Level DSA، لا يحتوى جذر DSE Superior Reference وبالتالي يكون نمط **supr**. وفي هذه الحالة، ولد Continuation Reference باستخدام معرفة عليا كما وجدت في جذر DSE. اضبط:

- **targetObject** على شيء مستهدف مبني من مكونات مستبانة باستخدام RDNs أولية (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات المتبقية غير المستبانة.
- **operationProgress.nameResolutionPhase** إلى **notStarted**؛
- **referenceType** إلى **superior**؛ و
- **accessPoints** as appropriate.

أضف Continuation Reference إلى قائمة Continuation References في **candidateRefs**. واستمر عند الخطوة 6.

4) تحقق إذا كانت العملية موجهة إلى مدخل جذر (?m = 0) . وإذا كانت، استمر عند الخطوة 5. وإذا لم تكن، ولد Continuation Reference باستخدام معرفة NSSR التي وجدت في جذر DSE. اضبط:

- **targetObject** على اسم شيء مستهدف مبني من مكونات مستبانة باستخدام RDNs أولية (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات المتبقية غير المستبانة.
- **operationProgress.nameResolutionPhase** إلى **proceeding**؛
- **operationProgress.nextRDNToBeResolved** إلى **1**؛
- **referenceType** إلى **nonSpecificSubordinate**؛ و
- **accessPoints** as appropriate.

أضف Continuation Reference إلى قائمة Continuation References في candidateRefs. واستمر عند الخطوة 6.

5) عند First Level DSA، يمكن فقط أداء عمليات List أو Search مع مدخل جذر كقاعدة شيء. ولهذا، إذا لم تكن العملية ليست List أو Search، يعاد **nameError** مع مشكلة **noSuchObject**. وإذا كانت عملية List أو Search، اضبط **nameResolutionPhase** على **completed** و يعاد مع **entry suitable**.

6) تحقق إذا كان هناك أي Continuation References في **candidateRefs**. وإذا كان **candidateRefs** فارغاً و**partialNameResolution** هو **FALSE**، Hعد **nameError** مع مشكلة **noSuchObject**. وإذا كان **candidateRefs** فارغاً و**partialNameResolution** هو **TRUE**، يضبط **partialName** على **TRUE** في النتيجة، و**nameResolutionPhase** على **completed**، ويعاد مع **entry suitable**. وإلا، استمر عند الخطوة 7.

7) استخدم وظيفة اختيار محلية لاختيار Continuation Reference من قائمة Continuation References في **candidateRefs**، أضفه إلى قائمة Continuation References في **NRcontinuationList** ويعاد مع **entry unsuitable**.

8) إذا لم يتمكن DSA من مواصلة Name Resolution (وفي أي حالة يكون **lastEntryFound** أقل من **nextRDNToBeResolved**)، استمر عند الخطوة 11. وإلا، استمر مع الخطوة التالية.

9) إذا كان DSE(i) هو DSE ظل مع معرفة تابع غير كاملة (يكون **subordinateCompletenessFlag** هو **FALSE**)، ولد Continuation Reference من نعت **supplierKnowledge** الذي وجد في **DSE(lastCP)**. اضبط:

- **targetObject** على اسم شيء مستهدف مبني من مكونات مستبانة باستخدام RDNs أولية (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات المتبقية غير المستبانة.
- **operationProgress.nameResolutionPhase** إلى **proceeding**؛
- **operationProgress.nextRDNToBeResolved** إلى **lastEntryFound**؛
- **referenceType** إلى **supplier**؛ و
- **accessPoints** as appropriate.

أضف Continuation Reference إلى قائمة Continuation References في **NRcontinuationList** ويعاد مع **entry unsuitable**.

10) إذا احتوي آخر مدخل وجد على NSSR (**DSE(lastEntryFound)** هو من نمط **nssr**)، ولد Continuation Reference من معرفة NSSR وجدت في **DSE(lastEntryFound)**. اضبط:

- **targetObject** على اسم شيء مستهدف مبني من مكونات مستبانة باستخدام RDNs أولية (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات المتبقية غير المستبانة.
- **operationProgress.nameResolutionPhase** إلى **proceeding**؛
- **operationProgress.nextRDNToBeResolved** إلى **lastEntryFound+1**؛
- **referenceType** إلى **nonSpecificSubordinate**؛ و
- **accessPoints** as appropriate.

أضف Continuation Reference إلى قائمة Continuation References في **candidateRefs**. واستمر عند الخطوة 7.

إذا كان **DSE(lastEntryFound)** ليس من نمط **nssr**، استمر عند الخطوة 6.

11) إذا كان **chainingArguments.referenceType** من نمط **nssr**، استمر عند الخطوة 13، وإلا عند الخطوة 12.

12) أعد **serviceError** مع مشكلة **invalidReference**.

13) إذا كان i +1 مساوي لـــ **nextRDNToBeResolved**، فإن الطلب قد تم تسييره هنا إلى NSSR وDSA غير قادر على مواصلة استبانة اسم؛ وفي هذه الحالة، اعد **serviceError** مع مشكلة **unableToProceed**، وإلا استمر عند الخطوة 12.

3.3.18 الإجراء الفرعي Target Found

يدخل الإجراء الفرعي هذا عندما يوائم اسم شيء مستهدف مدخل DSE محلي. ويتحقق الإجراء الفرعي هذا ما إذا كان المدخل الذي وجد مناسباً لمعالجة الطلب محلياً (كما يرد في الشكل 11):

1) استدعي إجراء Check Suitability.

2) إذا كان المدخل مناسباً (**entry suitable**)، عليك القيام بما يلي:

- اضبط **nameResolutionPhase** على **completed**.
- قارن قيمة **ChainingArguments.streamedResults** (إن وجدت) بعدد العناصر في **ChainingArguments.traceInformation**؛ وإذا كان مساوياً، اضبط StreamedResultOK على true؛
- أعد **entry suitable**.

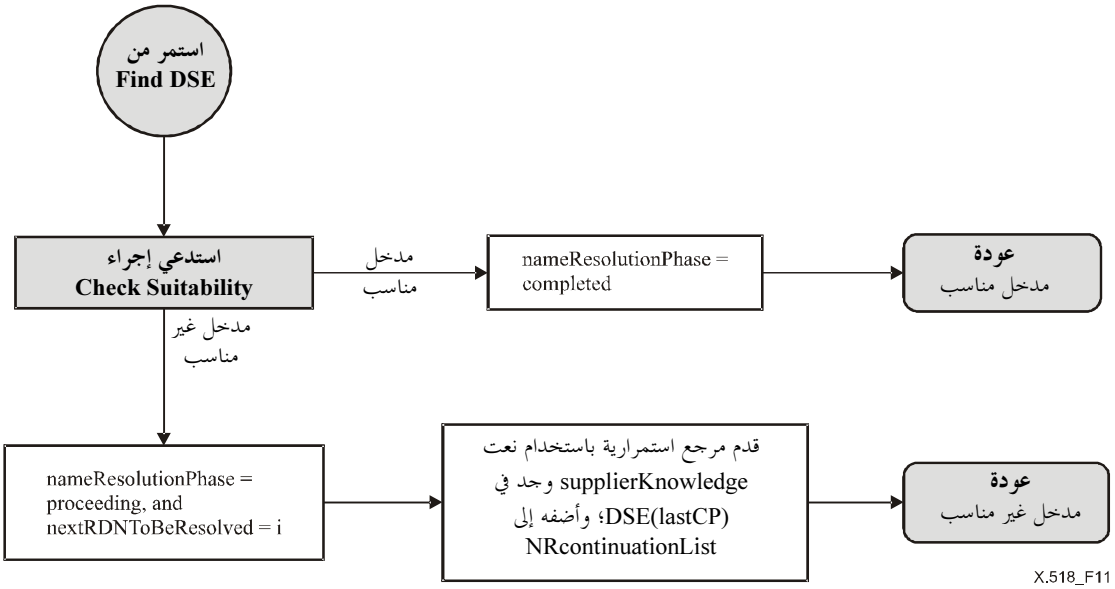
3) إذا كان المدخل غير مناسب (**entry unsuitable**)، ولد Continuation Reference باستخدام نعت **supplierKnowledge** الذي وجد في **DSE(lastCP)**. واضبط:

- **targetObject** على اسم شيء مستهدف مبني من مكونات مستبانة باستخدام RDNs أولية (يمكن تضمين قيم مميزة بديلة في RDNs) متسلسلة مع المكونات المتبقية غير المستبانة.
- **operationProgress.nameResolutionPhase** إلى **proceeding**؛
- **operationProgress.nextRDNToBeResolved** إلى **m**؛
- **referenceType** إلى **supplier**؛ و
- **accessPoints**.

أضف Continuation Reference إلى قائمة Continuation References في **NRcontinuationList** أعد **entry unsuitable**.

ملاحظة – إذا ضبطت مراقبة خدمة **localScope**، يمكن، مع ذلك، لـــ DSA، على أساس سياسات محلية، أن يقرر النظر في هذا المدخل على أنه مناسب ويستمر كما في الخطوة 2).

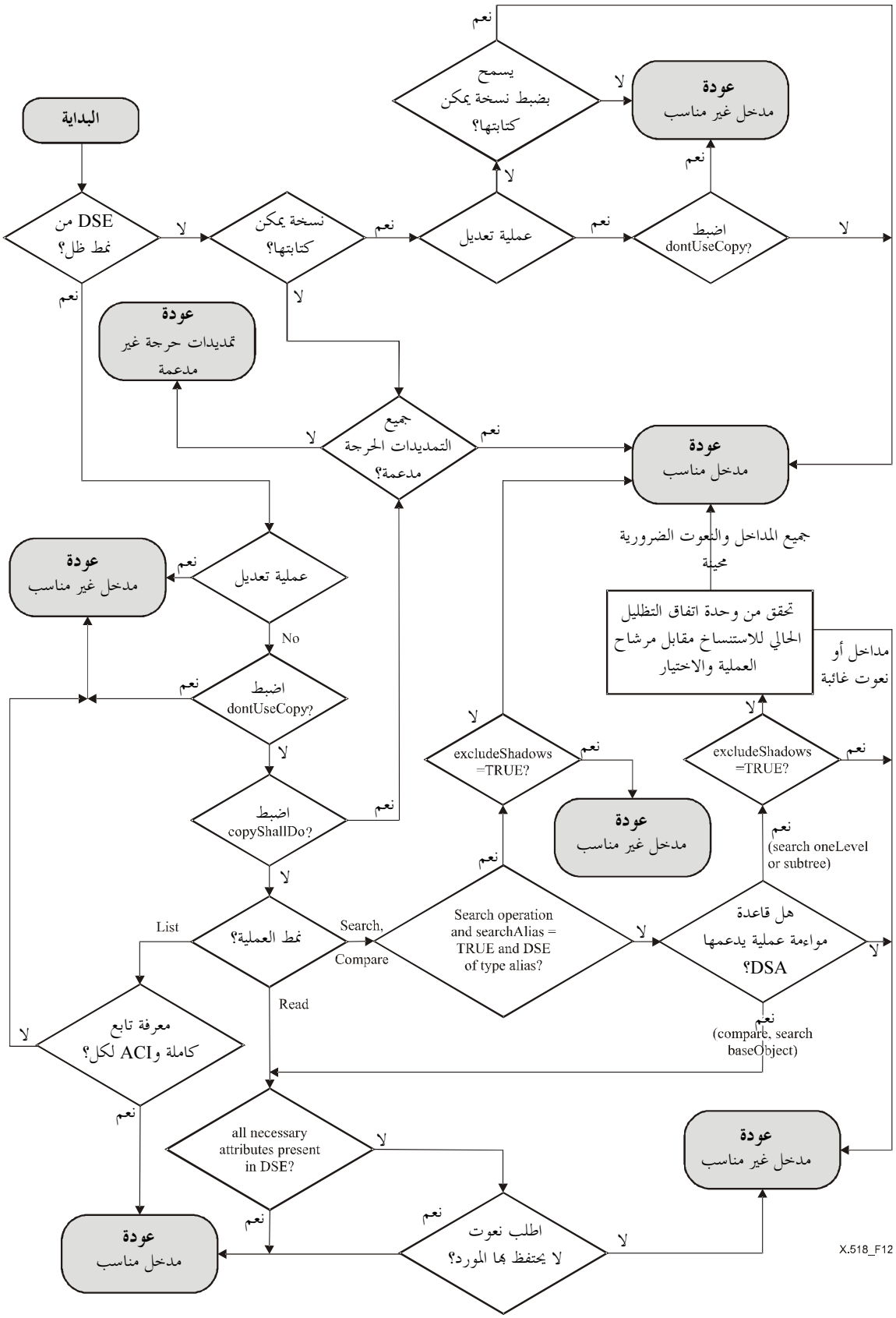
4) إذا لم يتم دعم تمديد حرج (**unsupported critical extension**)، يعاد **serviceError** مع مشكلة **unavailableCriticalExtension**.



الشكل 11 – الإجراء الفرعي Target Found

4.3.18 إجراء Check Suitability

يستدعي الإجراء هذا ليقرر ما إذا كان DSE الذي وجد مناسباً لأداء العملية المطلوبة (انظر الشكل 12). ويأخذ في الاعتبار **ChainingArguments** و**ServiceControls** والمتغيرات كما وردها المستعمل ونمط العملية وخواص DSE (ظل ومعرفة تابع ونعوت محينة وما إلى ذلك).



الشكل 12 – إجراء Check Suitability

1.4.3.18 معلمات إجراء

إن متغير الخرج لهذا الإجراء هو:

- مرجع لـــ DSE؛
- نمط العملية الذي يتحقق من مدى مناسبتها لـــ DSE؛
- **ChainingArguments**؛
- متغير العملية.

يكون الدخل إما **entry suitable** أو **entry unsuitable** أو **unsupported critical extension**.

1) إذا كان DSE ليس من نمط **shadow**، وليس من النمط **writeableCopy**، تحقق ما إذا كانت كل **criticalExtensions** يجرى دعمها. وإذا كانت مدعمة، عودة إلى مدخل مناسب، وإلا عودة إلى تمديدات حرجة غير مدعمة.

2) إذا كان DSE من نمط **shadow**. عودة إلى مدخل غير مناسب، إذا كان أي مما يلي true:

   - نمط العملية المطلوبة هو عملية تعديل.
   - مراقبة الخدمة **dontUseCopy** مضبوطة.

   وإلا استمر مع الخطوة التالية.

3) إذا كان DSE من نمط **writeableCopy**، عودة إلى مدخل غير مناسب، إذا كان أي مما يلي true:

   - نمط العملية المطلوبة هو عملية تعديل ومراقبة الخدمة **dontUseCopy**.مضبوطة.
   - نمط العملية المطلوبة هو عملية استفسار ومراقبة الخدمة **allowWriteableCopy** غير مضبوطة.

   وإلا اعد مدخل مناسب.

4) إذا ضبطت مراقبة الخدمة **copyShallDo**، تحقق إذا كانت جميع **criticalExtensions** مدعمة. وإذا كانت مدعمة، اعد **entry suitable**، وإلا عودة إلى **unsupported critical extension**.

5) إذا لم تضبط مراقبة الخدمة **copyShallDo**، تحقق إذا كانت جميع **criticalExtensions** مدعمة. وإذا كانت مدعمة، اذهب إلى الخطوة 5 وإلا عودة إلى **entry unsuitable**.

6) ميز بين أنماط العمليات:

   إذا كانت عملية List، استمر عند الخطوة 6.

   إذا كانت عملية Read، استمر عند الخطوة 7.

   إذا كانت عملية Search أو Compare، استمر عند الخطوة 8.

7) إذا كان للمدخل معرفة تابع كاملة، يمكن أداء عملية List. وفي هذه الحالة، عودة إلى **entry suitable**، وإلا عودة إلى **entry suitable**.

8) إذا كانت جميع النعوت المطلوبة محينة في DSE، عودة إلى **entry suitable**. وإذا كانت بعض النعوت غائبة، حدد بواسطة وسائل محلية ما إذا كانت نسخة ظل تحتفظ بجميع النعوت التي يحتفظ بها الرئيسي (مثلاً، بواسطة مرجع لاتفاق تظليل). وإذا كانت كذلك، يكون المدخل مناسباً (عودة إلى **entry suitable**). وإلا، قد يحتفظ المورد بالنعوت المطلوبة غير المحينة في الظل؛ وفي هذه الحالة، يتعين تسلسل الطلب (اعد **entry suitable**).

9) إذا كانت العملية هي **search** مع **searchAliases** مضبوطة على **TRUE** وDSE هو من نمط **alias** إذا كان **chainingArguments.excludeShadows** هو **FALSE** عودة إلى **entry suitable**، وإذا كان **TRUE** عودة إلى **entry unsuitable**.

10) إذا دعم DSA قاعدة المواءمة لـــ **compare** أو **search** كما طلب تكون العملية هي عملية **compare** أو **search** مع **subset** لـــ **baseObject**، ثم استمر عند الخطوة 7. إذا دعم DSA قاعدة المواءمة والعملية **search** مع مجموعة فرعية **oneLevel** أو **subtree**، استمر عند الخطوة 10. وإلا عودة إلى **entry unsuitable**.

11) إذا كان **chainingArguments.excludeShadows** هو **TRUE**، عودة إلى **entry unsuitable**. وإلا، تحقق من الفهم المحلي لمواصفة معلومات مظللة مقابل مرشاح واختيار العملية. وإذا كانت جميع المداخل والنعوت الضرورية محينة، عودة إلى **entry unsuitable**. وإذا كان أي مدخل أو نعت غائباً، عودة إلى **entry unsuitable**.

19 تقييم عملية

يعرف هذا القسم الإجراء الذي يتبعه DSA إذا كان المدخل المستهدف لعملية قد وجد محلياً (خلال Name Resolution). وطبقا لنمط عملية، ينفذ أحد الإجراءات التالية:

- لعملية **addEntry** أو **chainedAddEntry** أو **removeEntry** أو **chainedRemoveEntry** أو **modifyEntry** أو **chainedModifyEntry** أو **modifyDN** أو **chainedModifyDN** تتبع الإجراءات في 1.19.
- ولعملية **read** أو **chainedRead** أو **compare** أو **chainedCompare** تتبع الإجراءات في 2.19.
- ولعملية **search** أو **chainedSearch** أو **list** أو **chainedList** تتبع الإجراءات في 3.19.

1.19 إجراء تعديل

طبقاً لنمط عملية التعديل، تتبع الإجراءات المتطابقة في 1.1.19 إلى 4.1.19.

1.1.19 عملية Add Entry

1) يتحقق DSA من أن المرسل لديه حقوق نفاذ كافية، مثلاً، كالمعرفة في 5.1.11 من ITU-T X.511/ISO/IEC 9594-3. وإذا لم تكن، يعاد خطأ ملائم.

2) يضمن DSA أن مدخل مع اسم المدخل الذي يضاف لا يوجد. وإلا، يعيد **updateError** مع مشكلة **entryAlreadyExists**. وإذا كان DSE الأعلى هو من النوع الاضافي **nssr**، يتبع DSA الإجراء المعرف في 5.1.19 (عمليات Modify وNSSRs) لضمان أن اسم المدخل الجديد غير مبهم. وإذا شمل اسم المدخل الذي يضاف قيم متميزة متعددة متفاضلة بواسطة سياق لبعض النعوت في RDN النهائي، يضمن DSA لأي من RDNs البديلة الممكنة التي قد تبني ألا تؤدى (بغض النظر عن السياق) اسم لمدخل يوجد فعلاً.

3) إذا كان **targetSystem** محيناً، ولا تكون **AccessPoint** هي DSA الحالي، اذهب إلى الخطوة 4. إذا لم يكن **targetSystem** محيناً، أو محيناً، وتكون **AccessPoint** هي DSA الحالي، اذهب إلى الخطوة 5.

4) إذا كان المدخل هو مدخل فرعي، يعيد DSA **updateError** مع مشكلة **affectsMultipleDSAs**. وإذا لم يكن المدخل مدخلاً فرعياً، يكون لـــ DSA خيار محلي إذا كان يريد أم لا إنشاء HOB مع DSA محدد. وإذا لم يكن، يعيد DSA **serviceError** مع مشكلة **unwillingToPerform**، وإلا، ينشئ DSA إسناداً تشغيلياً تراتبياً مع DSA تابع محدد. وإذا دعم DOP، يتبع الإجراء في 1.1.13.24. وإلا، تستخدم وسائل محلية لانشاء HOB. وإذا لم يكن DSA على استعداد لانشاء إسناد تشغيلي، يعاد **serviceError** مع مشكلة **unwillingToPerform** لعملية **addEntry**. وإذا تم إنشاء HOB بنجاح، استمر عند الخطوة 7.

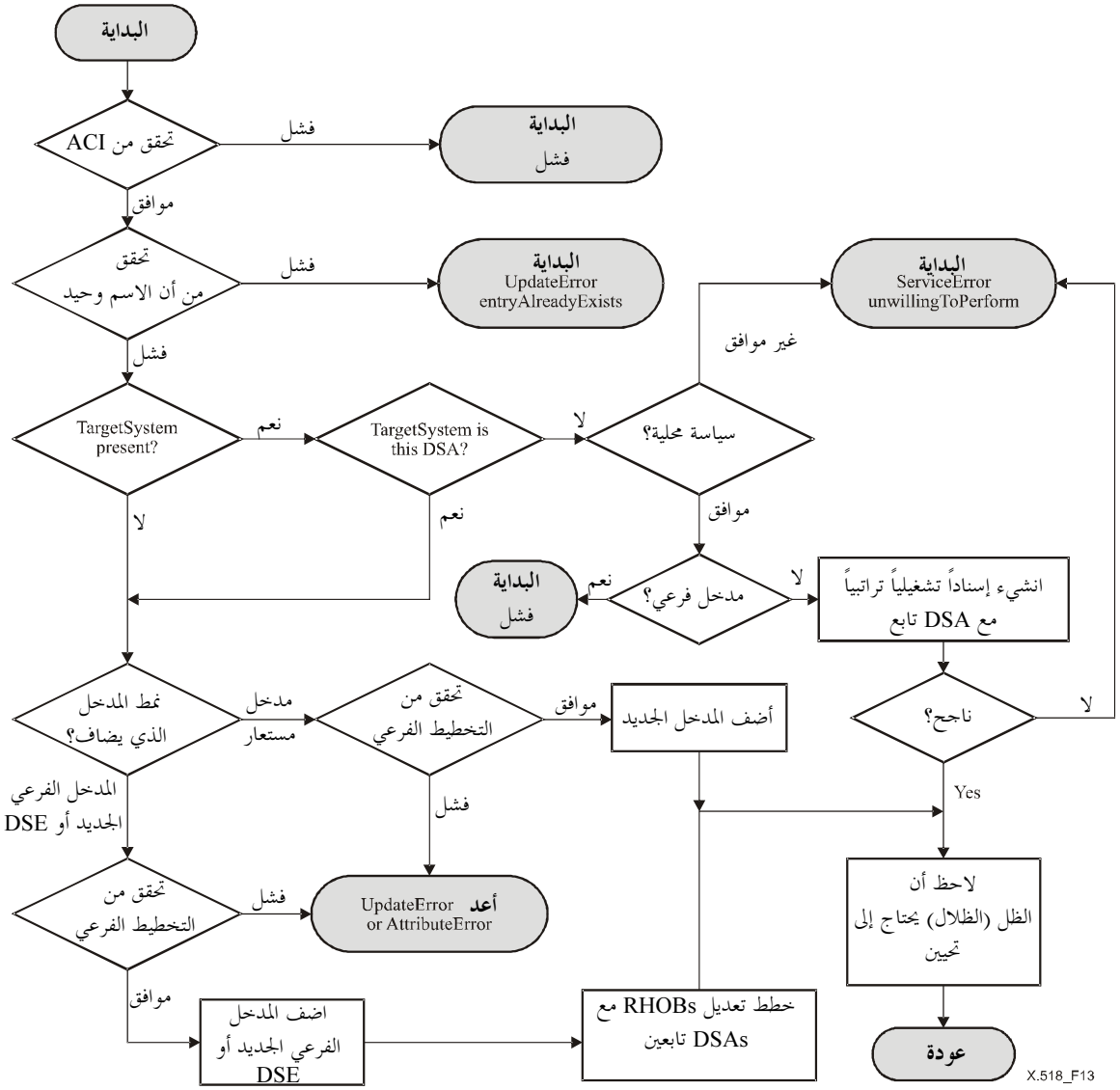
الملاحظة 1 – لا تنطبق الخطوة هذه من الإجراء على إنشاء مناطق إدارية مستقلة ذاتياً في DSA تابع.

5) يضمن DSA أن المدخل الجديد يتطابق مع التخطيط الفرعي، أو أن المدخل الفرعي الجديد أو DSE لأنماط أخرى يتطابق مع تخطيط النظام (مثلاً، أن DSE الأعلى المباشر لمدخل فرعي هو من نمط **admPoint**). وإذا لم يكن، يعيد **updateError** أو **attributeError**، وإلا يضيف DSE الجديد. وإذا كان مدخلاً، استمر عند الخطوة 7. وإذا كان مدخلاً فرعياً، استمر عند الخطوة 6. وإلا، تنفذ إجراءات إدارة معرفة ملائمة لأنماط أخرى لـــ DSE. انظر القسم 6.

6) يحيل DSA، في الوقت المناسب، عدل الإسناد التشغيلي لجميع DSAs التابعين يكون له إسنادات تراتبية أو تشغيلية تراتبية غير محددة. والإسنادات ذات العلاقة هي المتصاحبة مع سياقات تسمية تكون تابعة لـــ DSE الأعلى. ولا تكون سياقات التسمية التي تتطابق سابقاتها لسياق مع نقاط إدارية مستقلة ذاتياً ذات علاقة. وإذا دعم DOP، تتبع الإجراءات في 1.2.3.24 و2.3.25. وإذا لم يدعم DOP، تستخدم وسائل محلية لتعديل RHOBs.

الملاحظة 2 – يحدد إداري DSA الوقت المناسب، وقد يتراوح مباشرة بعد (أو حتى قبل) نتيجة العملية ويعاد إلى الاستراتيجية الدورية (مثلاً، عند ساعة محددة). وقد يتفاوت الوقت يعتمد على سبب التعديل، مثل، تحيينات ACI تنفذ مباشرة وتغير التخطيط الذي يتم دورياً.

7) إذا كان مدخل مضاف أو مدخل فرعي هو في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.



الشكل 13 – إجراء Add Entry

2.1.19 سحب عملية مدخل

1) يتحقق DSA من أن المبادر لديه حقوق نفاذ كافية، مثلاً، كما عرفت في 5.2.11 من ITU-T X.511/ISO/IEC 9594-3. وإذا لم يكن، يعاد خطأ مناسب.

2) يضمن DSA أن المدخل الذي يسحب هو مدخل ورقة. وإلا، يعيد DSA **updateError** مع مشكلة **.notAllowedOnNonLeaf**

3) يتم التحقق من نمط DSE الذي يسحب. وإذا كان **subentry**، استمر عند الخطوة 5. ,إذا كان **cp**، استمر عند الخطوة 6. وإذا كان **entry** أو **alias**، استمر عند الخطوة 4. وإلا، تنفذ إجراءات إدارة معرفة ملائمة لأنماط أخرى لـــ DSE. انظر القسم 6.

4) اسحب المدخل أو المدخل المستعار واستمر عند الخطوة 7.

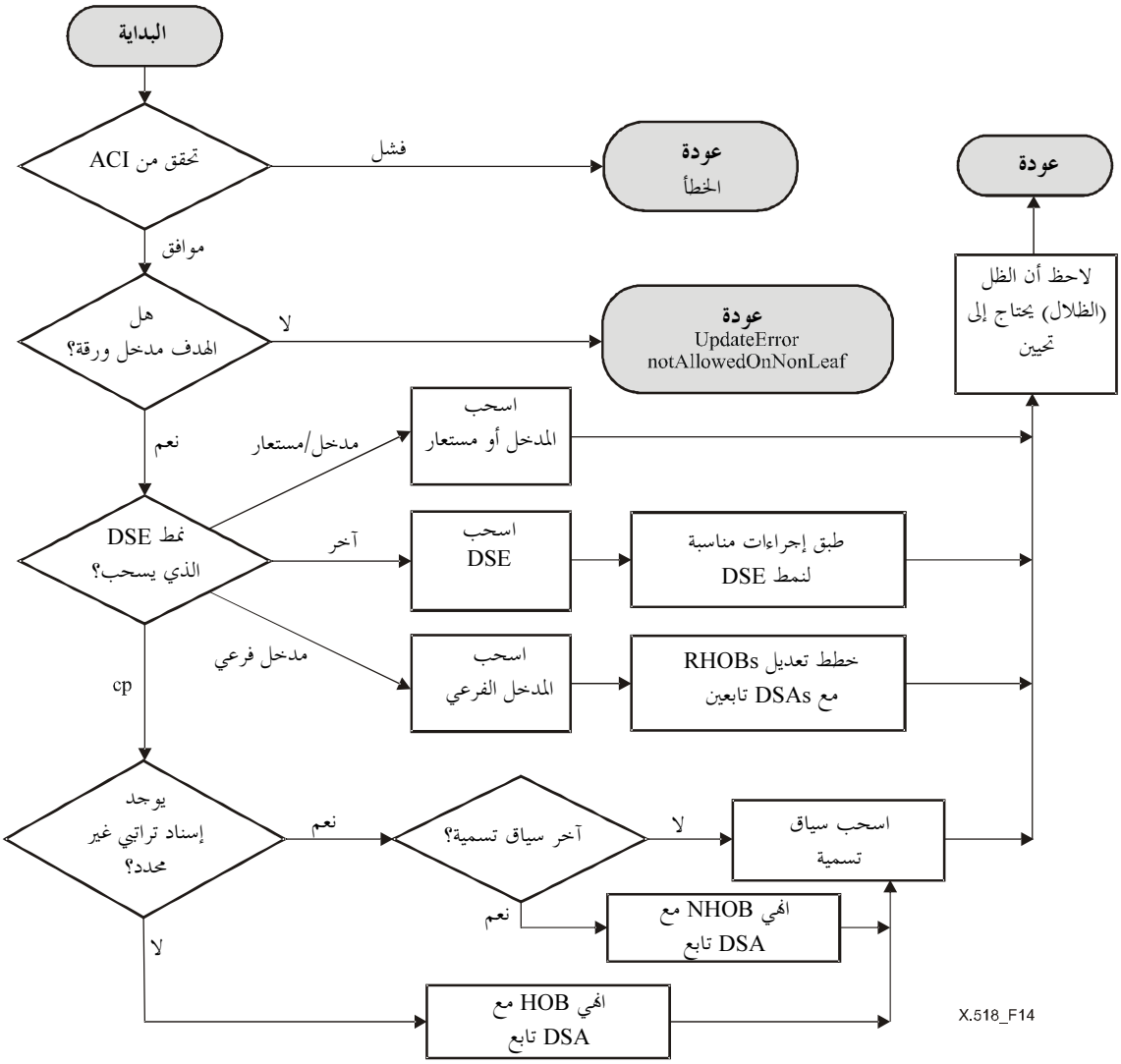
5) اسحب المدخل الفرعي. في الوقت المناسب، عدل الإسناد التشغيلي لجميع DSAs التابعين ليكون لـــ DSA إسنادات تراتبية أو تشغيلية تراتبية غير محددة. والإسنادات ذات العلاقة هي المتصاحبة مع سياقات تسمية تابعة لـــ DSE الأعلى.

ولا تكون سياقات التسمية التي تتطابق سابقاتها لسياق مع نقاط إدارية مستقلة ذاتيا ذات علاقة. وإذا دعم DOP، تتبع الإجراءات في 1.2.3.24 و2.3.25. وإلا، تستخدم وسائل محلية. واستمر عند الخطوة 7.

6) اسحب سياق التسمية. وإذا كان لدى DSA إسناد تشغيلي تراتبي لسياق التسمية هذا، ينهي الإسناد التشغيلي التراتبي مع DSA الأعلى المباشر. وإذا كان لـــ DSA إسناد تشغيلي تراتبي غير محدد لسياق التسمية هذا، وهذا آخر سياق تسمية لإسناد تشغيلي تراتبي غير محدد، ينهي الإسناد التشغيلي التراتبي غير المحدد مع DSA الأعلى المباشر. وإذا دعم DOP، تتبع الإجراءات في 2.3.3.24 و2.3.3.25. وإلا، تستخدم وسائل محلية لإنهاء RHOB.

7) إذا كان سياق التسمية المسحوب مدخل أو مدخل مستعار أو مدخل فرعي في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر، يجرى تحيين مستهلكي الظل المستهلكين باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.

إذا كان مرجع مسحوب أو مرجع تابع غير محدد في DSA عالٍ مباشر (الذي تم إنهاء RHOB) في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.



الشكل 14 – سحب إجراء مدخل

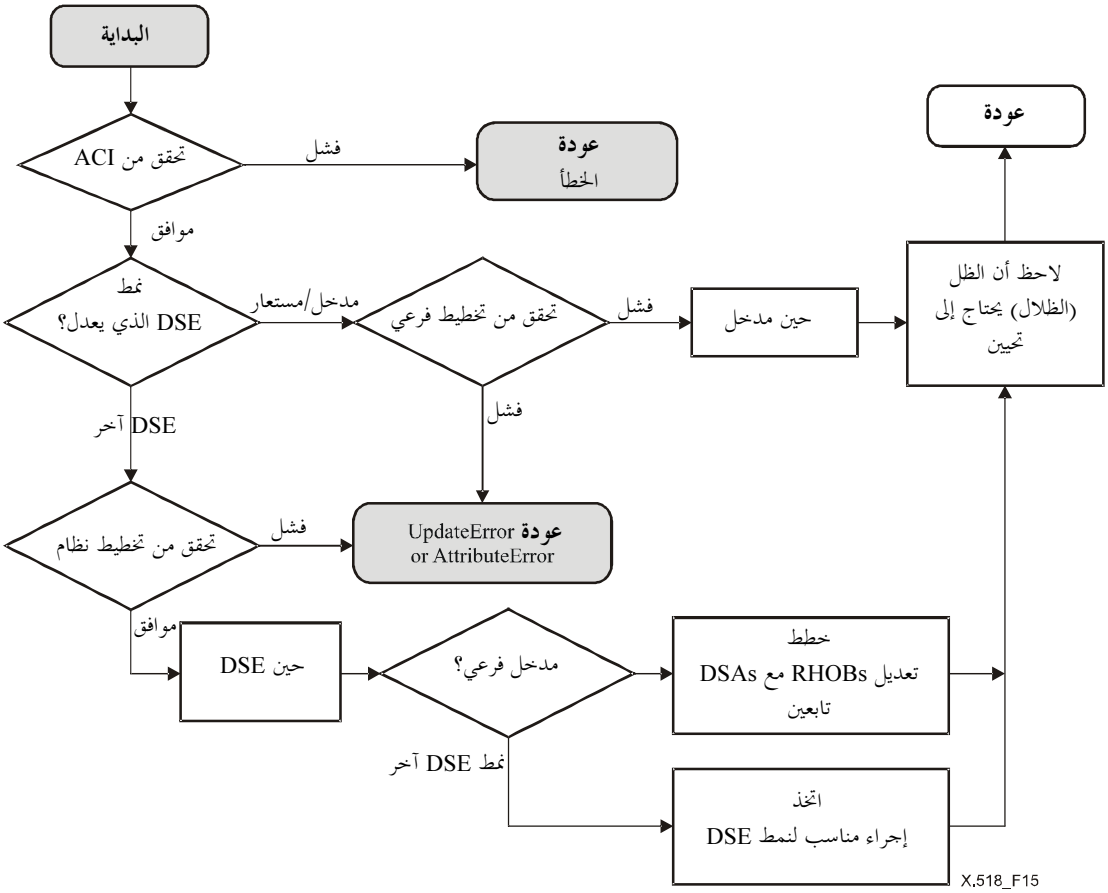
3.1.19 تعديل عملية مدخل

1) يتحقق DSA من أن المبادر لديه حقوق نفاذ كافية، مثلاً، كما عرفت في 5.3.11 من ITU-T X.511/ISO/IEC 9594-3. وإذا لم يكن، يعاد خطأ مناسب.

2) تتطابق التعديلات على المدخل أو المستعار مع التخطيط الفرعي. ويتطابق التعديل مع DSE لأنماط أخرى، بما في ذلك مدخل فرعي، مع تخطيط النظام. وإلا، يعيد DSA **updateError** أو **attributeError** مناسب. وبعد أداء التعديلات، إذا كان DSE المستهدف هو من نمط **subentry**، استمر عند الخطوة 3؛ وإذا كان DSE المستهدف هو من نمط **entry** أو **alias**، استمر عند الخطوة 4. وإلاّ، تنفذ إجراءات إدارة معرفة ملائمة لأنماط أخرى لـــ DSE. انظر القسم 6

3) يعدل DSA، في الوقت المناسب، الإسناد التشغيلي لجميع DSAs التابعين ليكون له إسنادات تراتبية أو تشغيلية تراتبية غير محددة. والإسنادات ذات العلاقة هي المتصاحبة مع سياقات تسمية تابعة لنقطة إدارية حيث المدخل الفرعي المعدل يوجد أسفله. ولا تكون سياقات التسمية التي تتطابق سابقاتها لسياق مع نقاط إدارية مستقلة ذاتياً ذات علاقة. وإذا دعم DOP، تتبع الإجراءات في 1.2.3.24 و2.3.25. وإلا، تستخدم وسائل محلية.

4) إذا كان المدخل أو مدخل مستعار أو مدخل فرعي المعدل في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.



الشكل 15 – إجراء تعديل مدخل

4.1.19 عدل عملية DN

1) يتحقق DSA من أن المبادر لديه حقوق نفاذ كافية، مثلاً، كما عرفت في 5.3.11 من ITU-T X.511/ISO/IEC 9594-3. وإذا لم يكن، يعاد خطأ مناسب.

2) إذا كانت العملية إما حرك مدخل أو حرك مدخل وغير اسمه المميز النسبي، اذهب إلى الخطوة 3. وإذا كانت العملية هي لتغيير الاسم المميز النسبي فقط لمدخل، اذهب إلى الخطوة 4.

3) تؤدى العملية طبقا للتعريف في 1.4.11 من ITU-T X.511/ISO/IEC 9594-3. وإذا لم يكن الرئيسي القديم أو الرئيسي الجديد أو المدخل أو أي من توابعه في DSA هذا، أو إذا كان للرئيسي الجديد NSSRs، ترفض العملية مع **updateError** مع مشكلة **affectsMultipleDSAs**. ويضمن DSA ألا يوجد مدخل آخر مع الاسم الجديد. وإلا، يعاد **updateError** مع مشكلة **entryAlreadyExists**. ويضمن DSA أن الاسم الجديد للمدخل يتطابق مع التخطيط الفرعي. وإلاّ، يعيد **attributeError** أو **updateError**. وإذا لم تظهر أي من هذه المشاكل عليك أن تحرك المدخل (تغيير RDN إذا طلب ذلك) واذهب إلى الخطوة 9.

4) يطبق النص التالي على تغيير اسم مميز نسبي لمدخل، الذي قد يكون أو لا يكون مدخل ورقة، والذي يكون له أو لا يكون له تابع واحد أو أكثر في DSA واحد أو أكثر. ويجرى التحقق من نمط DSA للمدخل الذي يعاد تسميته. وإذا كان **subentry**، استمر عند الخطوة 7. وإذا كان **cp**، استمر عند الخطوة 6. وإذا كان **entry** أو **alias**، استمر عند الخطوة 5.

5) يضمن DSA ألا يوجد مدخل آخر مع الاسم الجديد. وإلا، يعاد **updateError** مع مشكلة **entryAlreadyExists**. وإذا كان DSE الأعلى هو من النوع الاضافي **nssr**، يتبع DSA الإجراء المعرف في 5.1.19 (عمليات Modify وNSSRs) لضمان أن اسم المدخل الجديد غير مبهم. وإذا شمل اسم المدخل الذي يضاف قيم متميزة متعددة متفاضلة بواسطة سياق لبعض النعوت في RDN النهائي، يضمن DSA لأي من RDNs البديلة الممكنة التي قد تبني ألا تؤدى (بغض النظر عن السياق) اسم لمدخل يوجد فعلاً. ويضمن DSA أن الاسم الجديد للمدخل يتطابق مع التخطيط الفرعي. وإلا، يعيد **attributeError** أو **updateError**. أعد تسمية المدخل أو المدخل المستعار. وإذا كان المدخل هو مدخل غير ورقة وله توابع في DSAs آخرين، استمر عند الخطوة 8، وإلا استمر عند الخطوة 9.

6) يضمن DSA أن الاسم الجديد للمدخل يتطابق مع التخطيط الفرعي. وإلاّ، يعيد **attributeError** أو **updateError** مناسب.

إذا كان DSA له HOB مع DSA أعلى، يحاول DSA التابع أن يعدل HOB قبل الاستجابة لعملية Modify DN. ويضمن DSA الأعلى ألا يوجد مدخل آخر مع الاسم الجديد، قبل قبول التعديل. وإذا دعم DOP، تتبع الإجراءات في 2.2.3.24. وإذا لم يدعم DOP، فهي مسألة محلية لكيفية تعديل HOB ويجرى التحقق من الاسم الجديد من أجل تفرده. وإذا عدل HOB بنجاح، وكان لسياق تسمية سياقات تسمية تابعة في DSAs آخرين، اذهب إلى الخطوة 8؛ وإلا، اذهب إلى الخطوة 9. وإذا لم يكن من الممكن تعديل HOB يعاد **updateError** مع مشكلة **affectsMultipleDSAs**.

إذا كان DSA له NHOB لسياق التسمية هذه مع DSA أعلى، تعتبر مسألة كيفية اكتشاف استنساخ مداخل خارج مدى مواصفة الدليل هذا. ويعاد تسمية المدخل. وإذا كان لسياق تسمية سياقات تسمية تابعة في DSAs آخرين، اذهب إلى الخطوة 8؛ وإلا، اذهب إلى الخطوة 9.

7) ويضمن DSA أن الاسم الجديد للمدخل الفرعي يتطابق مع تخطيط نظام. وإلا، يعيد **attributeError** أو **updateError** مناسب. يضمن DSA ألا يوجد مدخل فرعي آخر مع الاسم الجديد. وإلا، يعيد **updateError** مع مشكلة **entryAlreadyExists**.

8) يعدل DSA، في الوقت المناسب، الإسناد التشغيلي لجميع DSAs التابعين ليكون له إسنادات تراتبية أو تشغيلية تراتبية غير محددة. والإسنادات ذات العلاقة هي المتصاحبة مع جميع سياقات تسمية تابعة للمدخل الذي يجرى إعادة تسميته أو لسياقات التسمية التابعة لنقطة إدارية مستقلة ذاتياً يكون مدخلها الفرعي قد أعيد تسميته. وسياقات التسمية التي تتطابق سابقات تسميتها مع نقاط إدارية مستقلة ذاتياً هي غير ذي علاقة. وإذا دعم DOP، تتبع الإجراءات في 1.2.3.24 و2.3.25. وإلاّ، تستخدم وسائل محلية لتعديل RHOBs.

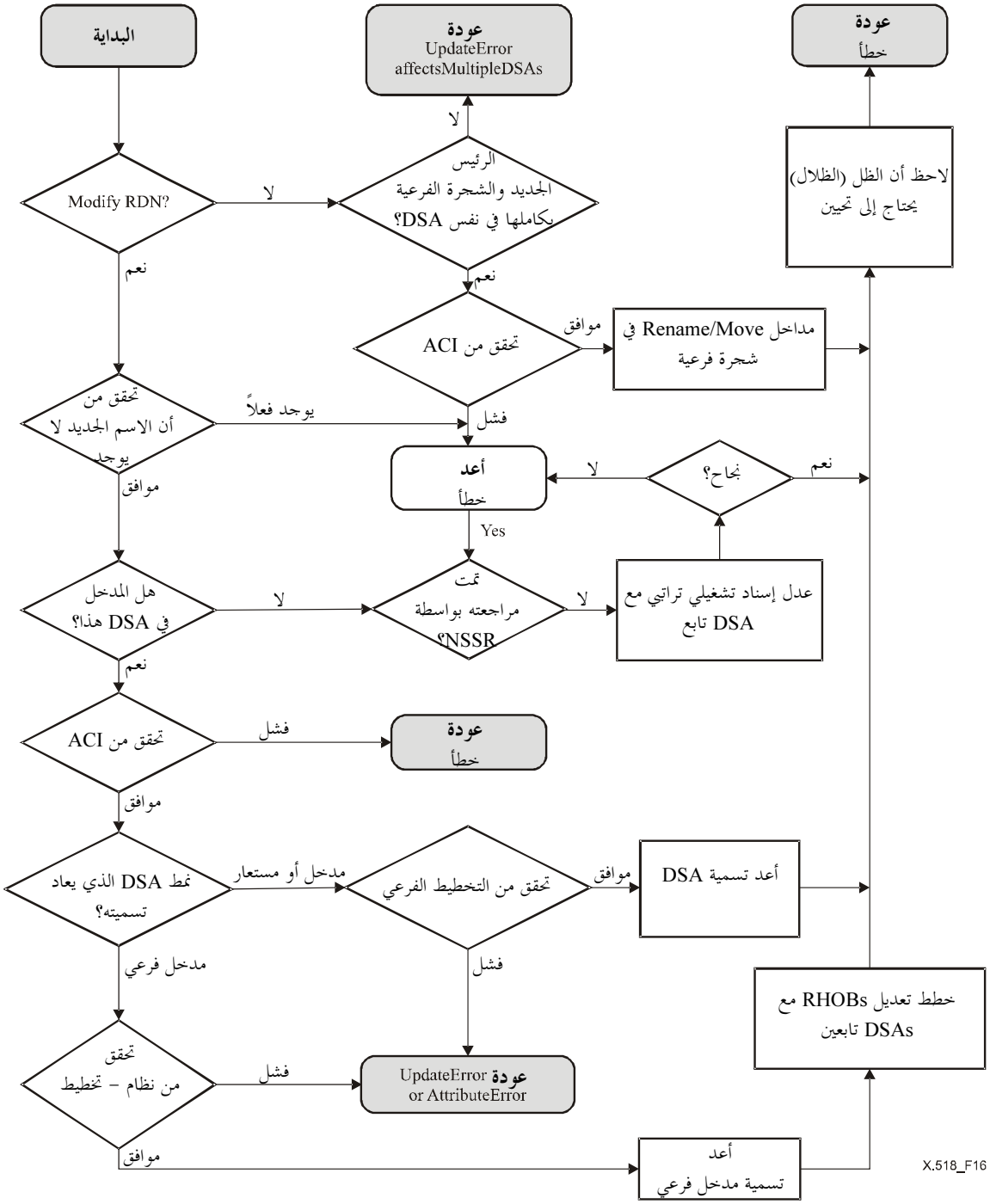
9) إذا كان سياق التسمية المعاد تسميته أو المدخل أو أي من تابعيه أو مدخل مستعار أو مدخل فرعي في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر يحتفظ به DSA، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.

وإذا كان المدخل أو المدخل المستعار أو مدخل فرعي في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر يحتفظ به DSA، ولم يكن الرئيسي لمدخل أو مدخل مستعار أو مدخل فرعي معاد تسميته في **UnitOfReplication**، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-؛ وفي هذه الحالة، يسحب المدخل المظلل وجميع توابعه.

وإذا كان المدخل أو المدخل المستعار أو مدخل فرعي ليس في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر يحتفظ به DSA، يكون المدخل أو المدخل المستعار أو مدخل فرعي معاد تسميته هو الآن في **UnitOfReplication**، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-؛ وفي هذه الحالة، يظلل المدخل المظلل وجميع توابعه.

إذا كان المرجع التابع المعاد تسميته في DSA الأعلى المباشر (الذي عدل HOB في الخطوة 6 أعلاه) في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-.

إذا كانت مكونات RHOB مع DSA تابع (كما عدل في الخطوة 8) في **UnitOfReplication** لاتفاق تظليل واحد أو أكثر يحتفظ به DSA تابع، يجرى تحيين مستهلكي الظل باستخدام إجراءات خدمة ظل معلومات الدليل الواردة في ITU-T X.525/ISO/IEC 9594-9.



الشكل 16 – إجراء Modify DN

5.1.19 عمليات Modify و Non-Specific Subordinate References

إذا كان DSA له NSSRs ولا يعرف المجموعة الكاملة لأسماء توابع مدخل، الذي إما:

أ) عملية **addEntry** قد تم توجيهها؛ أو

ب) عملية **modifyDN** قد تم توجيهها،

فيمكن لـــ DSA أن يؤدى المجموعة التالية من الإجراءات قبل أداء العملية.

1) إذا ضبط خيار مراقبة خدمة **chainingProhibited** على **addEntry** أو **modifyDN**، اعد **updateError** مع مشكلة **affectsMultipleDSAs**.

2) إذا كان DSA غير مستعد أو غير قادر على تسلسل متعدد لطلبات جارية، اعد **serviceError** مع مشكلة **unwillingToPerform** أو **unavailable**، على التوالي.

3) يقوم DSA بتسلسل متعدد لعملية **chainedReadEntry** لكل DSA رئيسي في مجموعة **accessPointInformation** لـــ NSSR. (يستخدم DSA فقط DSA الرئيسي من كل **MasterAndShadowAccessPoints** الواجب العبور بعدم اتساق تسبب فيه التظليل). وتضبط معلمات **ReadArgument** كما يلي:

object إما لاسم المدخل الذي يضاف (في حالة **addEntry**) أو لاسم مقترح لمدخل موجود (في حالة **modifyDN**).

selection نعت صنف شيء.

تضبط معلمات **CommonArguments** كما يلي:

- اضبط خيار مراقبة خدمة **dontDereferenceAliases**.
- اضبط **OperationProgress.nameResolutionPhase** على **completed**.

تضبط معلمات **ChainingArguments** كما يلي:

- اضبط **originator** على اسم المرسل؛
- يشطب **targetObject**؛
- اضبط **OperationProgress.nameResolutionPhase** على **proceeding** و**nextRDNToBeResolved** على (عدد RDNs في اسم شيء) −1.
- اضبط **traceInformation** على تتابع فارغ؛
- اضبط **referenceType** على **nonSpecificSubordinate**؛
- **timeLimit**، حسب الاقتضاء طبقاً للطلب الواصل.

المعلمات الأخرى، مثل، **SecurityParameters**، يمكن أن تضبط حسب الاقتضاء، مثلاً، بواسطة سياسة محلية.

4) ينتظر DSA اكتمال مجموعة الاستجابات. وإذا كانت إحدى الاستجابات هي **ReadResult**، يعاد خطأ كما في (6) أدناه.

5) إذا كانت جميع الاستجابات هي **serviceError** مع مشكلة **unableToProceed**، يمكن مواصلة تقييم عملية.

6) إذا أعيد **ReadResult**، يعاد **updateError** مع مشكلة **entryAlreadyExists** إلى العملية الأصلية.

7) إذا أعيد أي خطأ آخر إلى طلب **readEntry**، يعاد **serviceError** مع مشكلة **unwillingToPerform**.

يعطي DSA المستقبل لـــ **chainedRead** استجابة طبقاً لوجود او عدم وجود المدخل، وسياسة مراقبة نفاذه.

2.19 إجراء استفسار مدخل وحيد

تقع عمليات **read** و**chainedRead** و**compare** و**chainedCompare** في زمرة إجراءات استفسار مدخل وحيد. وتحتوى هذه الإجراءات فقط على الخطوات الثلاث التالية:

1) تحقق من مراقبة نفاذ، كما ورد في القسم 9 من ISO/IEC 9594-3 | ITU-T X.511. وإذا لم يسمح بالعملية، أعد خطأ أمن مناسب.

2) أدى العملية على DSE الذي وجد كما في القسم 9 من ISO/IEC 9594-3 | ITU-T X.511.

3) قم بإعداد الاجابة، وعودة.

3.19 إجراء استفسار مدخل متعدد

طبقاً لنمط عملية الاستفسار (**list** أو **search**)، تتبع الإجراءات المتطابقة المعرفة في 1.3.19 و2.3.19.

1.3.19 إجراءات List

يحدد القسم الفرعي هذا إجراء التقييم المحدد لعمليات **list** و**chainedList**.

يتبع الإجراء **List (I)** عندما يضبط مكون **operationProgress.nameResolutionPhase** لطلب List على **notStarted** أو **proceeding** وعندما يجد DSA، بعد أداء Name Resolution، أنه يحتفظ بقاعدة شيء. ويتبع الإجراء **List (II)** عندما يضبط مكون **nameResolutionPhase** لطلب List على **completed**.

1.1.3.19 معلمات إجراء

1.1.1.3.19 متغيرات

تكون المتغيرات التي يستخدمها هذا الإجراء هي:

- the **ListArgument**;
- the target DSE **e**;
- **operationProgress** of the **chainingArgument**.

2.1.1.3.19 نتائج

إذا نفذ الإجراء هذا بنجاح، يعيد:

- مجموعة من التوابع لـــ **e** في **listInfo.subordinates**؛
- **limitProblem** مشار إليها في **partialOutcomeQualifier**؛
- مجموعة من مراجع استمرارية في **SRcontinuationList**.

2.1.3.19 تعريف إجراء

1.2.1.3.19 إجراء List (I)

يتألف إجراء **List (I)** من الخطوات التالية الواردة في الشكل 17:

1) إذا ضبطت مراقبة الخدمة **subentry**، اذهب إلى الخطوة 5؛ وإلا، اذهب إلى الخطوة 17؛

2) إذا كان **e** DSE من نمط **nssr**، أضف Continuation Reference إلى **SRcontinuationList** مع المكونات التالية:

- **targetObject** إلى اسم مميز أولي لـــ **e** DSE (قد يتضمن قيم مميزة بديلة في RDNs)؛
- **aliasedRDNs** غائبة؛
- **operationProgress** مع **nameResolutionPhase** مضبوطة على **completed** و**nextRDNtoBeResolved** غائب؛
- **rdnsResolved** غائب؛
- **referenceType** مضبوط على **nonSpecificSubordinate**؛
- **accessPoints** مضبوطة على **accessPointInformation** تشتق كل واحدة من قيمة نعت **nonSpecificKnowledge** لـــ **e** DSE.

3) ولكل **e** DSE لتابع مباشر لـــ **e** DSE، تنفذ الخطوات التالية:

أ) تحقق من أن ACI في e متاحاً. وإذا لم يسمح ACI بـــ RDN listing لـــ e، عليك تخطي DSE هذا. وإذا لم يكن ACI متاحا (مثلاً في حالة مراجع تابعة وglue)، فهي سياسة محلية ما إذا تواصلت.

ب) تحقق من جميع أنماط DSE لـــ e.

'1' إذا كان **e** من نمط **subr**، توجد حالتان. وفي الحالة الأولى، المدخل التابع لـــ ACI وصنف شيء متاح محلياً، وعلى أي حال، قائم على أساس سياسة محلية وسماح لـــ ACI، أضف RDN لـــ **e**، إلى **listInfo.subordinates** مع **aliasEntry** مضبوط على **TRUE** إذا كان **e** من نمط **sa**، و**fromEntry** مضبوط على **FALSE**. والحالة الأخرى هي عندما يكون ACI للمدخل غير متاح في **e** وعلى أي حال، أضف Continuation Reference إلى **SRcontinuationList** مع المكونات التالية:

- **targetObject** إلى اسم مميز أولي لـــ **e** DSE (قد يتضمن قيم مميزة بديلة في RDNs)؛
- **aliasedRDNs** غائبة؛
- **operationProgress** مع **nameResolutionPhase** مضبوطة على **completed** و**nextRDNtoBeResolved** غائب؛
- **rdnsResolved** غائب؛
- **referenceType** مضبوط على **subordinate**؛
- **accessPoints** مضبوطة على القيمة الواردة في **specificKnowledge** لـــ **e** DSE.

'2' إذا كان **e** من نمط **entry** أو **glue**، أضف RDN لـــ **e'**، إلى **listInfo.subordinates** مع **aliasEntry** مضبوط على **FALSE** و**fromEntry** مضبوطاً طبقاً إذا كان **e'** نسخة.

ملاحظة – في حالة أن **e'** هو **glue**، ينبغي أن يكون له تابع واحد أو أكثر يعني أنه لا يمكن أن يكون مستعاراً في DSA. وأيضاً، يخزن أي ACI يتعلق بعملية List في DSE هذا، مورد عبر بروتوكول تظليل.

'3' إذا كان DSE **e** من نمط **alias**، أضف RDN لـــ **e'**، إلى **listInfo.subordinates** مع **aliasEntry** مضبوط على **TRUE** و**fromEntry** مضبوطاً طبقاً إذا كان **e'** نسخة

ج) تحقق إذا كان حد الوقت أو الحجم أو الإداري قد تم تجاوزه. وإذا كان كذلك، اضبط **limitProblem** بناء على ذلك في **partialOutcomeQualifier** وعودة.

د) استمر من الخطوة 3 حتى تعالج جميع DSEs التابعة.

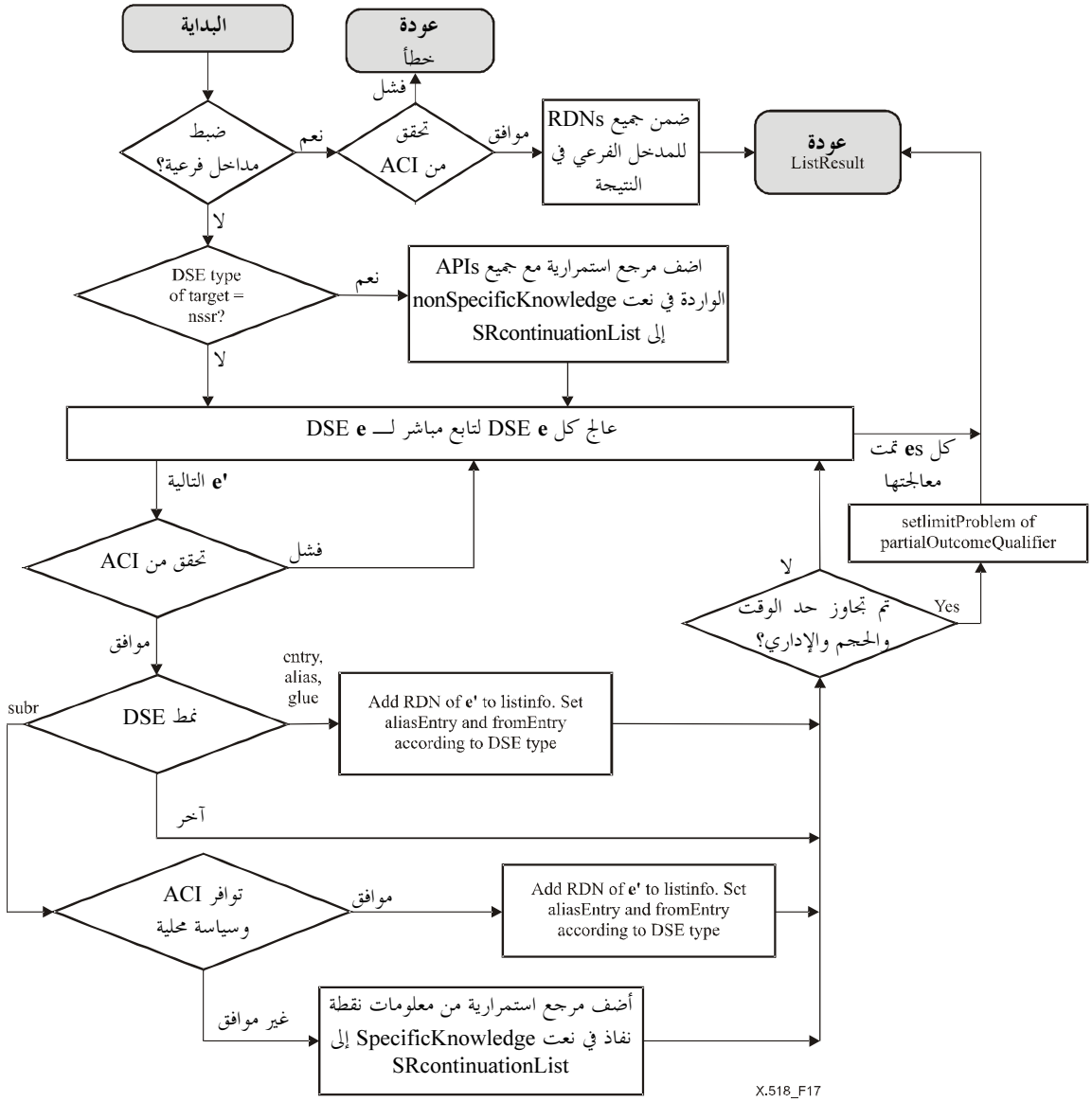
4) إذا تمت معالجة جميع DSEs التابعة، عودة إلى **Operation Dispatcher**.

5) تنفذ الخطوات التالية لكل مدخل فرعي **e'** تابع مباشرة DSE .0e

أ) تحقق من أن ACI في e متاحاً. وإذا لم يسمح ACI بـــ RDN listing لـــ **e'**، عليك تخطي DSE هذا. وإلاّ، أضف RDN لـــ **e'**، إلى **listInfo.subordinates** مع **aliasEntry** مضبوطاً على **FALSE** و**fromEntry** مضبوطاً طبقاً إذا كان **e'** نسخة.

ب) تحقق إذا كان حد الوقت أو الحجم أو الإداري قد تم تجاوزه. وإذا كان كذلك، اضبط **limitProblem** بناء على ذلك في **partialOutcomeQualifier** وعودة إلى.

6) وعودة إلى **Operation Dispatcher**.



الشكل 17 – إجراء List (I)

2.2.1.3.19 إجراء List (II)

يتألف إجراء **List (II)** من الخطوات التالية الواردة في الشكل 18:

1) لكل DSEs **e'** لتابع مباشر لـــ DSE **e**، نفذ الخطوات من 1 أ) إلى 1 د):

أ) إذا لم يكن **e** مدخل أو مستعار، استمر مع التابع المباشر التالي.

ب) تحقق من ACI لـــ **e**. وإذا لم يسمح ACI للعملية، استمر مع التابع المباشر التالي لـــ **e**.

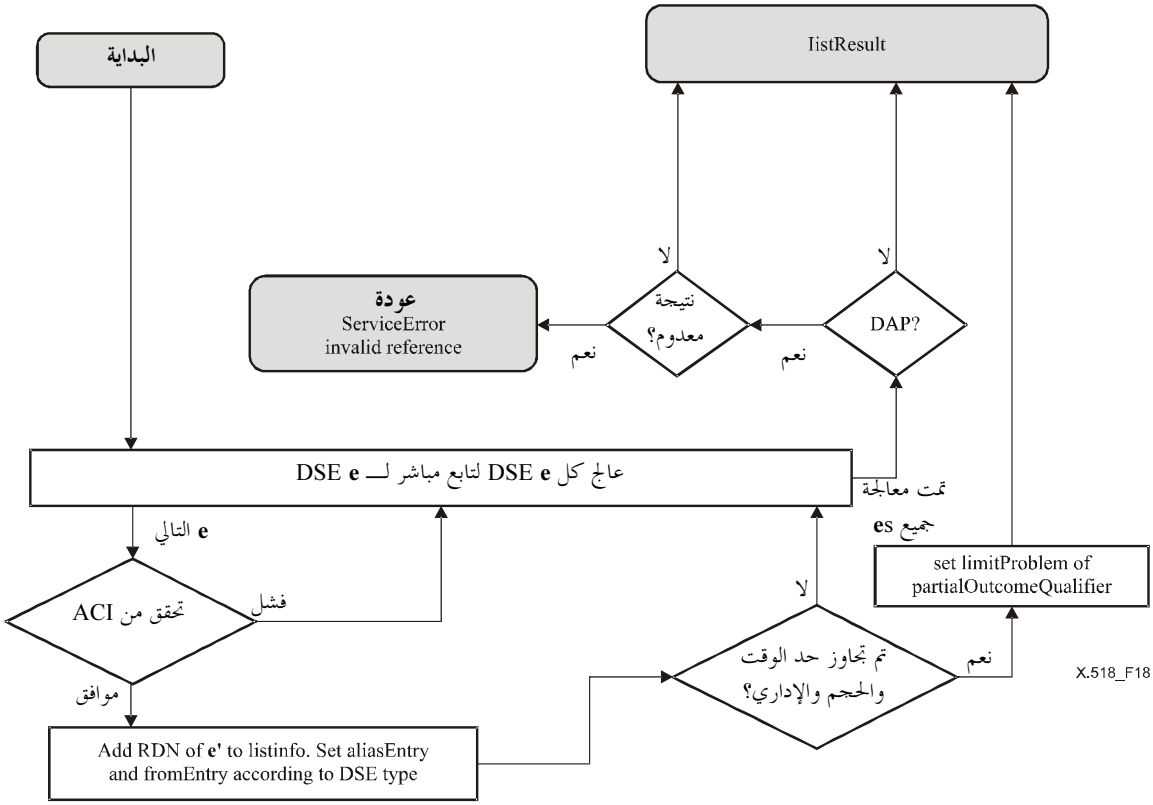
ج) أضف RDN لـــ DSE **e** إلى **listInfo.subordinates** مع مكون **aliasEntry** لـــ **listInfo.subordinates** إذا كان **e** مستعار ومكون **fromEntry** مضبوطاً يعتمد ما إذا كان **e** نسخة أم لا. وتجاهل DSEs من نمط **shadow** أو **writableCopy** إذا كان **excludeShadows** هو **TRUE**.

د) تحقق إذا كان حد الوقت أو الحجم أو الإداري قد تم تجاوزه. وإذا كان كذلك، اضبط **limitProblem** لـــ **partialOutcomeQualifier** طبقاً لذلك وأعد.

هـ) استمر من الخطوة 1 أ) حتى تعالج جميع DSEs التابعة.

2) إذا تمت معالجة جميع DSEs التابعة، تحقق إذا كان الطلب الفرعي هذا جاء من DAP أو DSP. وفي حالة تقديم الطلب الفرعي هذا عبر DAP.، تكون **ListResult** فارغة، ثم عودة إلى **serviceError** مع مشكلة **invalidReference** إلى Operation Dispatcher. وإلا، عودة إلى **ListResult**.

ملاحظة – يستخدم **invalidReference** كتحذير أمني في حالة عدم وجود نفاذ للمستعمل إلى المدخل الأعلى. وإذا كان المدخل الأعلى ACI متاحاً (موفر بواسطة RHOB)، قد تعاد نتيجة معدوم إذا سمح بذلك.



الشكل 18 – إجراء List (II)

2.3.19 إجراءات Search

يحدد القسم الفرعي هذا إجراء التقييم المحدد لعمليات **search** و**chainedSearch**.

يتبع الإجراء **Search-rule-check (I)** عندما يضبط مكون **operationProgress.nameResolutionPhase** لطلب Search على **notStarted** أو **proceeding** وعندما يجد DSA، بعد أداء Name Resolution، أنه يحتفظ بشيء مستهدف. وإذا أعاد هذا الإجراء خطأ، عد مع ذلك الخطأ. وإلا، يتبع إجراء **Search (I)**.

يتبع الإجراء **Search-rule-check (II)** عندما يضبط مكون **nameResolutionPhase**. لطلب **search** على **completed**. وإذا أعاد هذا الإجراء خطأ، عد مع ذلك الخطأ. وإلا، يتبع إجراء **Search (II)**.

ملاحظة – عندما يكون **nameResolutionPhase** هو **completed**، يتوقع أن يكون الشيء المستهدف رئيسي مباشر لسابقة سياق.

1.2.3.19 معلمات إجراء

1.1.2.3.19 متغيرات

إن المتغيرات التي يستخدمها هذا الإجراء هي:

- the **SearchArgument**;
- the target DSE **e**;
- **operationProgress** of the **ChainingArguments**;
- **exclusions** of the **ChainingArguments** (a list of RDNs to exclude from search);
- **traceInformation** of the **ChainingArguments**;
- **searchRuleId** of the **ChainingArguments**;
- **chainedRelaxation** of the **ChainingArguments**; and
- **relatedEntry** of the **ChainingArguments**.

2.1.2.3.19 نتائج

إذا نفذ الإجراء هذا بنجاح، يعيد:

- مجموعة من مداخل متوائمة في **searchResult.entryInformation**؛
- **alreadySearched** في **ChainingResults**؛
- يعتمد على الشروط، عد في **partialOutcomeQualifier.entryCount**؛
- مجموعة من مراجع استمرارية في **SRcontinuationList**.

2.2.3.19 تعريف إجراء

1.2.2.3.19 إجراء Related Entry Argument

يكون هذا الإجراء متعلقاً إذا كان طلب **search** له مكون **joinArguments** و**ChainingArguments** (إن وجد) وليس له مكون **relatedEntry**.

1) إذا كان طلب **search** محمياً، ولد طلب DSP لكل عنصر لمكون **joinArguments** متضمن كل منه في طلب DAP الأصلي أو LDAPMessage. وتتبع **ChainingArguments** كما يلي:

- إذا كان الطلب الواصل له **ChainingArguments** مع مكون **originator**، تنسخ قيمة هذا المكون في مكون **originator** للطلبات المولدة؛ وإلاّ، فتحدد استخدام هذا المكون سياسة الأمن المحلية.

  ملاحظة – قد لا يتمكن DSA المستقبل من الاستفادة من الاسم الذي يعطي للمكون هذا، نظراً لأنه من DIT منفصلة.

- يشطب مكون **operationProgress** أو يضبط على قيمة بالتغيب؛
- تشطب مكونات **traceInformation** و**aliasDereferenced** و**aliasedRDNs** و**returnCrossRefs** و**entryOnly** و**exclusions** و**nameResolutionOnMaster** و**searchRuleId** و**chainedRelaxation**؛
- يضبط مكون **relatedEntry** على قيمة متطابقة مع موضع نسبي لـــ **JoinArgument** الذي ينطبق على DSA الذي يحال إليه الطلب؛ وفي حالة إعطاء أول **JoinArgument** القيمة صفر، تكون القيمة التالية واحد وما إلى ذلك.

2) إذا لم يكن طلب Search الواصل محمياً، ولد طلب DSP لكل عنصر لمكون **jointArguments** حيث يولد **SearchArgument** كما يلي:

- ينسخ مكون **baseObject** من مكون **joinBaseObject** لـــ **JoinArgument** المتطابق؛

- ينسخ مكون **subset** من مكون **joinSubset** لـــ **JoinArgument** المتطابق؛
- ينسخ مكون **filter** من مكون **filter** لـــ **JoinArgument** المتطابق؛
- تكون المكونات الباقية كما في الطلب الأصلي، باستثناء أن مكونات **joinArguments** و**joinType** تشطب.

تكون **ChainingArguments** كما هي أعلاه للطلبات المحمية، باستثناء أن مكون **relatedEntry** يشطب.

3) استدعي **Operation Dispatcher** لكل طلب يستمر محلياً.

4) إذا أعاد **Operation Dispatcher** خطأ **referral** أو مشغول أو أخطاء غير متاحة ثم يضيف (أو يقدم ويضيف) مرجع استمرارية إلى **partialOutcomeQualifier** لـــ **SearchResult**، ويعيد.

5) إذا أعاد **Operation Dispatcher** أخطاء أخرى، عليك تجاهله، وعودة.

6) إذا أعاد **Operation Dispatcher SearchResult**، فإن:

'1' إذا وقعت النتيجة أو جفرت، أو وقعت وجفرت، اضفها إلى **uncorrelatedSearchInfo** في **SearchResult**.

'2' إذا لم توقع النتيجة أو جفرت، أو لم توقع وجفرت، أدى عملية join كما ورد في ITU-T X.511/ISO/IEC 9594-3.

2.2.2.3.19 إجراء Search-rule-check (I)

يتعلق هذا الإجراء فقط إذا دعم DSA مناطق إدارة محددة لخدمة.

إذا كان مكون **searchRuleId** محيناً في **ChainingArguments**، تكون العملية هي نتيجة إجراء تبديل مستعار خلال مرحلة التقييم السابقة. وإذا كان DSE المستهدف في المنطقة الإدارية المحددة لخدمة له **dmdId** مختلف أو كان DSE المستهدف خارج المنطقة الإدارية المحددة لخدمة، يعاد خطأ خدمة مع **unwillingToPerform**. وإلاّ، اختار search-ruled قائم على معلومات في **searchRuleId** وعودة.

الملاحظة 1 – عرفت إدارة خدمة على أنها تمديد حرج. وعندما يستقبل DSA، الذي لا يدعم إدارة خدمة، طلب بحث متسلسل مع مكون **searchRuled**، يعيد **serviceError** مع مشكلة **unavailableCriticalExtension**.

إذا لم يكن **searchRuleId** محيناً وكان DSE المستهدف خارج المنطقة الإدارية المحددة لخدمة؛ أو في مثل هذه المنطقة، ولكن لا تتصاحب مداخل فرعية مع تلك المنطقة، وعودة.

إذا كان DSE المستهدف في المنطقة الإدارية المحددة لخدمة وتكشف **traceInformation** عن أن العملية كانت في مرحلة تقييم سابقة، اعد مع خطأ خدمة **unwillingToPerform**.

الملاحظة 2 – هذا هو الحال حيث بدأ بحث تقييمه الإبتدائي خارج المنطقة الإدارية المحددة لخدمة ويحاول الآن الانتشار في منطقة إدارية محددة لخدمة مختلفة.

وإلا، يتبع الإجراء التالي:

1) حدد موقع جميع search-rules المتصاحبة مع DSE المستهدف، أي، جميع search-rules في مداخل فرعية لخدمة لها DSE مستهدف في مواصفات شجرته الفرعية (مثلاً، باستخدام نعت تشغيلي **searchRulesSubentry**). وتسمي search-rules فيما يلي *candidate-search-rules*. وإذا لم توجد

search-rules هذه، ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**، لتشمل مكون **notification** لنعت **searchServiceProblem** a **CommonResults** مع القيمة **id-pr-unidentifiedOperation**، واعد.

2) إذا شمل **serviceType** و/أو مراقبة خدمة **userClass** في طلب بحث، احذف جميع search-rules التي لا تمتثل لمراقبة النفاذ تلك من candidate-search-rules. وإذا ترك ذلك القائمة فارغة، ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**؛ وضمن في مكون **notification** لـــ **CommonResults** المعلومات كما وردت بالتفصيل أدناه، واعد:

- نعت **searchServiceProblem** مع قيمة **id-pr-unidentifiedOperation**؛
- إذا كانت مراقبة خدمة **serviceType** شاملة في طلب بحث، نعت **serviceType** مع القيمة لمراقبة الخدمة تلك.

3) قسم قائمة candidate-search-rule إلى أربع قوائم (قد يكون بعضها فارغاً):

- قائمة **GoodPermittedSR** المحتوية على جميع candidate-search-rules التي نفذ الطالب لها سماح والتي يمتثل لها طلب بحث طبقاً لإجراء search-validation الوارد في ITU-T X.511/ISO/IEC 9594-3.

ملاحظة 3 – إذا كانت هذه القائمة فارغة، ليس هناك سبب لوضع قوائم أخرى.

- قائمة **MatchProblemSR** المحتوية على جميع candidate-search-rules التي نفذ الطالب لها سماح والتي يمتثل لها طلب بحث باستثناء matchingUse في request-attribute-profiles واحد أو أكثر؛
- قائمة **BadPermittedSR** المحتوية على جميع candidate-search-rules التي نفذ الطالب لها سماح ولكن لا يمتثل لها طلب بحث؛
- قائمة **DeniedSR** المحتوية على جميع candidate-search-rules التي لم ينفذ الطالب لها سماح.

4) إذا كانت قائمة **GoodPermittedSR** تحتوي على search-rule فارغة واحدة أو أكثر، اختار باستخدام خوارزمية محلية لـــ search-rules الفارغة هذه باعتبارها إدارة search-rule وعودة.

5) إذا لم تكن قائمة **GoodPermittedSR** فارغة، تجاهل جميع search-rules باستثناء ذات دلالة **userClass** الأعلى.

6) اختار search-rule المتبقية في قائمة **GoodPermittedSR** باعتبارها إدارة search-rule، باستخدام خوارزمية محلية، وعودة.

ملاحظة 4 – إذا وجدت في القائمة أعلاه search-rules عديدة لتختار منها، ينبغي أن يسجل التنفيذ الحدث للاهتمام الإداري، لأن من المحتمل أن تحتاج تعاريف search-rule إلى إعادة العمل بشأنها.

7) إذا لم تكن قائمة **MatchProblemSR** فارغة، اختار واحدة من search-rules متتبعاً خوارزمية مماثلة للمحددة في 5 و6 أعلاه، وولد خطأ خدمة ومعلومات متصاحبة كما ورد في 13-4 من ITU-T X.511/ISO/IEC 9594-3، ثم عودة.

8) إذا كانت قائمة **DeniedSR** فارغة، استمر مع 10؛ وإلا، تجاهل أي search-rule من القائمة التي لا يمتثل لها طلب بحث وتجاهل أي search-rule فارغة. وإذا كانت القائمة فارغة الآن، استمر مع 10؛ وإلا، ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**؛ وضمن في مكون **notification** لـــ **CommonResults** المكونات الفرعية كما وردت بالتفصيل أدناه، وعودة:

- نعت **searchServiceProblem** مع قيمة **id-pr-unavailableOperation**؛
- إذا كانت search-rules المتبقية في قائمة **DeniedSR** لها نفس القيمة في مكون **serviceType**، نعت **serviceType** مع تلك القيمة.

9) إذا كان **BadPermittedSR** فارغاً، ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**؛ وضمن في مكون **notification** لـــ **CommonResults** المكونات الفرعية كما وردت بالتفصيل أدناه، وعودة:

- نعت **searchServiceProblem** مع قيمة **id-pr-unidentifiedOperation**.

10) لكل بند مرقم في الإجراء في 13-1 من ITU-T X.511/ISO/IEC 9594-3 حسب الترتيب، تحقق من طلب بحث مقابل search-rules المتبقية في **BadPermittedSR**، ثم لكل بند:

- إذا امتثل البحث للبند لبعض search-rules، ولكن ليس لجميع search-rules، تجاهل قواعد البحث التي لا تمتثل؛
- إذا احتفظ **BadPermittedSR** الآن فقط بـــ search-rule واحدة، ادى الإجراء المحدد في القسم 13 من ITU-T X.511/ISO/IEC 9594-3، وعودة؛
- وإلا، يجرى التحقق من البند التالي.

11) إذا احتفظ **BadPermittedSR** الآن فقط بـــ search-rules بحيث لا يمتثل للبحث طبقا للإجراء حتى الآن، ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**؛ وضمن في مكون **notification** لـــ **CommonResults** المكونات الفرعية كما وردت بالتفصيل أدناه، وعودة:

- نعت **searchServiceProblem** مع قيمة **id-pr-unidentifiedOperation**.
- إذا حددت جميع search-rules في **BadPermittedSR** نفس service-type، يكون نعت **serviceType** مع service-type ذلك كقيمة.

12) لكل بند مرقم في الإجراء في 13-2 من ITU-T X.511/ISO/IEC 9594-3 حسب الترتيب، تحقق من طلب بحث مقابل search-rules المتبقية في **BadPermittedSR**، ثم لكل بند:

- إذا امتثل البحث للبند لبعض search-rules، ولكن ليس لجميع search-rules، تجاهل قواعد البحث التي لا تمتثل؛
- إذا احتفظ **BadPermittedSR** الآن فقط بـــ search-rule واحدة، ادى الإجراء المحدد في القسم 13 من ITU-T X.511 | ISO/IEC 9594-3، وعودة؛
- وإلا، يجرى التحقق من البند التالي.

13) لكل بند مرقم في الإجراء في 13-3 من ITU-T X.511/ISO/IEC 9594-3 حسب الترتيب، تحقق من طلب بحث مقابل search-rules المتبقية في **BadPermittedSR**، ثم لكل بند:

- إذا امتثل البحث للبند لبعض search-rules، ولكن ليس لجميع search-rules، تجاهل قواعد البحث التي لا تمتثل؛
- إذا احتفظ **BadPermittedSR** الآن فقط بـــ search-rule واحدة، ادى الإجراء المحدد في القسم 13 من ITU-T X.511 | ISO/IEC 9594-3، وعودة؛

- وإلا، يجرى التحقق من البند التالي.

14) ولد خطأ خدمة مع مشكلة **requestedServiceNotAvailable**؛ ضمن في مكون **notification** لـــ **CommonResults** المكونات الفرعية كما وردت بالتفصيل أدناه، وعودة:

- نعت **searchServiceProblem** مع قيمة **id-pr-unidentifiedOperation**؛

- إذا حددت جميع search-rules في **BadPermittedSR** نفس service-type، يكون نعت **serviceType** مع service-type ذلك كقيمة.

3.2.2.3.19 إجراء Search-rule check (II)

يتعلق هذا الإجراء فقط إذا دعم DSA مناطق إدارية محددة لخدمة.

إذا كان **searchRuleId** غير محين، وجميع المداخل التابعة مباشرة (سابقات سياق) لـــ DSE المستهدف هي نقاط إدارية محددة لخدمة، اعد **serviceError** مع مشكلة **unwillingToPerform**. ومع ذلك، إذا كانت بعض المداخل التابعة ليست نقاط إدارية محددة لخدمة، اختار سياقات تسمية متوافقة لكل تقييم بحث وعودة.

إذا كان **searchRuleId** محيناً، يجرى التحقق من كل مدخل تابع لـــ DSE المستهدف للتحقق من أنه في نفس منطقة إدارة محددة لخدمة مثل DSE المستهدف. وإذا لم يكن، بستثني سياق تسمية متوافق من البحث. وإذا كانت هناك سياقات تسمية متبقية (بما في ذلك سياقات في DSA المنفذ) حيث يمكن استمرار البحث، اختار search-rule المعرفة في **searchRuleId** واعد. وإذا لم تكن هناك سياقات تسمية متبقية يمكن أن يستمر فيها البحث، ولد **serviceError** مع مشكلة **unwillingToPerform** وعودة.

ملاحظة – ينبغي ألا يحث الأخير إذا كانت معلومات معرفة متسقة بين DSA وDSA محتفظ بسياق تسمية أعلى.

4.2.2.3.19 اختيار معلومات مدخل

بالنسبة لمداخل متوائمة ومدخل مختار كجزء من اختيار تراتبي، تختار معلومات نعت كتقاطع:

أ) لما يحدده **searchArgument.selection**، من الممكن معدل بواسطة مواصفات سياق بالتغيب، ولمداخل متوائمة بواسطة **searchArgument.matchedValuesOnly**؛

ب) لما تحدده governing-search-rule (إن وجد).

تضاف معلومات هذا المدخل إلى قائمة مداخل في **searchResult.entryInformation**.

أضف فقط نعوت حجمها (نمط وجميع القيم) ليس أكبر من **attributeSizeLimit**.

5.2.2.3.19 إجراء Search (I)

هذا إجراء تكراري ينطبق على طلب **search** الذي يبدأ عند مدخل مستهدف **e**. ويبحث المدخل المستهدف **e** ويعالج التابع المباشر لـــ DSEs لـــ **e**. وينفذ الإجراء نفسه تكرارياً في حالة أن الشجرة الفرعية بكاملها يجرى بحثها. ويتألف الإجراء من الخطوات التالية كما ترد في الشكل 19.

1) إذا كان نمط **e** DSE هو نمط **cp** (DSE عند سابقة سياق)، تحقق إذا كان أي عنصر لمتغير **exclusions** هو سابقة DN لـــ **e**.

أ) إذا كان كذلك، عودة.

ب) وإلا، استدعي Check Suitability.

'1' إذا كان **e** غير مناسب، قدم **continuationReference** كما يلي واضفه إلى **SRContinuationList**:

- يضبط **targetObject** على DN لـــ DSE **e**؛
- يضبط **operationProgress** مع **nameResolutionPhase** على **proceeding** ويضبط **nextRDNtoBeResolved** على عدد RDNs في e .
- ولا تتغير جميع مكونات **continuationReference**.

ثم اعد.

الملاحظة 1 – هذا هو المكان الوحيد الذي يتسلسل عنده الطلب الفرعي search لمورد الظل. وبمعنى آخر، يكون دائماً الشيء المستهدف للطلب الفرعي المتسلسل سابقة سياق.

'2' وإلا، أضف اسم مميز لـــ **e** إلى **alreadySearched** في **ChainingResults**.

الملاحظة 2 – تحتوي **alreadySearched** على سابقات سياق فقط.

2) إذا كان **e** من نمط **alias** و**searchAliases** في **SearchArgument** هو **TRUE** استدعي إجراء **Search Alias** ثم اعد.

3) إذا كان **subset** هو **oneLevel**، استمر مع الخطوة 6.

ملاحظة 3 – لا يمكن لـــ **e** أن يكون تابع غير مكتمل عند هذه النقطة نظراً لأن Check Suitability عند سابقة سياق ينبغي أن تؤكد أن هذا لا يمكن أن يحدث.

4) إذا كان **subset** هو **baseObject** أو إذا كان **entryOnly** هو **TRUE** استمر مع هذه الخطوة؛ وإلا، اذهب إلى الخطوة 5.

إذا كان واحداً مما يلي true:

أ) **e** من نمط **subentry** تضبط مراقبة الخدمة **subentry**؛ أو

ب) **e** ليس من نمط **subentry** لا تضبط مراقبة الخدمة **subentry**، عليك القيام بالخطوات التالية:

'1' تحقق من ACI. وإذا لم يسمح بالعملية، اعد.

'2' طبق متغير المرشاح المحدد في **SearchArgument.filter** على **e** DSE. واضمن أن النفاذ إلى جميع النعوت المستخدمة في المرشاح مسموح بها كما ورد في ITU-T X.501/ISO/IEC 9594-2. فإذا تواءم المرشاح وإذا لم يستثن المدخل نتيجة لاختيار تراتبي، أضف معلومات النعت كما وردت في 3.2.2.3.19.

'3' إذا تضمنت مراقبة بحث **hierarchySelection** في طلب **search** (من الممكن معدل بواسطة مواصفة search-rule)، يكون المدخل جزء من زمرة تراتبية لها أكثر من عضو واحد، وتضبط أكثر من دلالة **self**، واستدعي إجراء **Hierarchy Selection (I)**.

ثم اعد.

5) إذا كان **subset** هو **subtree** (و **entryOnly** ليس **TRUE**)، بالإضافة إلى واحد مما يلي هو true:

أ) **e** من نمط **subentry** تضبط مراقبة الخدمة **subentry**؛ أو

ب) **e** ليس من نمط **subentry** لا تضبط مراقبة الخدمة **subentry**، عليك القيام بالخطوات التالية:

'1' تحقق من ACI. وإذا لم يسمح بالعملية، اذهب إلى الخطوة 6.

'2' طبق متغير المرشاح المحدد في **SearchArgument.filter** على **e** DSE. واضمن أن النفاذ إلى جميع النعوت المستخدمة في المرشاح مسموح بها كما ورد في ITU-T X.501/ISO/IEC 9594-2. فإذا تواءم المرشاح وإذا لم يستثن المدخل نتيجة لاختيار تراتبي، أضف معلومات النعت كما وردت في 3.2.2.3.19.

'3' إذا تضمنت مراقبة بحث **hierarchySelection** في طلب **search** (من الممكن معدل بواسطة مواصفة search-rule)، يكون المدخل جزء من زمرة تراتبية لها أكثر من عضو واحد، وتضبط أكثر من دلالة **self**، واستدعي إجراء **Hierarchy Selection (I)**.

'4' واصل الخطوة 6.

6) إذا كان **e** من نمط **nssr**، أضف Continuation Reference إلى **SRcontinuationList** مع المكونات التالية:

- **targetObject** إلى اسم مميز أولي لـــ **e** DSE (قد يتضمن قيم مميزة بديلة في RDNs)؛
- **aliasedRDNs** غائبة؛
- **operationProgress** مع **nameResolutionPhase** مضبوطة على **completed** و**nextRDNtoBeResolved** غائب؛
- **rdnsResolved** غائب؛
- **referenceType** مضبوط على **nssr**؛
- **accessPoints** مضبوطة على **AccessPointInformation** مشتقة من القيم الموجودة في نعت **nonSpecificKnowledge**.

7) عالج جميع **e'** DSE التي حددت مواقعها مباشرة لتابع **e** DSE المستهدف حتى تتم معالجة DSEs التابعين. وإذا كان **e** في منطقة إدارية محددة لخدمة، تجرى معالجة DSEs التابعين مباشرة الذين هم جزء من نفس المنطقة الإدارية المحددة لخدمة. وإذا كان **e** خارج منطقة إدارية محددة لخدمة، لا تعالج DSEs التابعين مباشرة الذين هم جزء من نفس المنطقة الإدارية المحددة لخدمة. وخلال هذه العروة، وإذا تجاوزت قائمة المداخل المتوائمة في **searchResult.entryInformation** حد الحجم أو تجاوزت حد الوقت أو الحد الإداري تضبط على **limitProblem** طبقاً لذلك في **partialOutcomeQualifier** واعد.

الملاحظة 4 – يطبق ضمنياً التحقق من حد الحجم في كل مرة يحين **searchResult**.

أ) إذا كان **e'** من نمط **subr**، وليس من نمط **cp**، ولا يمثل مدخل تابع يكون نقطة إدارية محددة لخدمة، أضف Continuation Reference إلى **SRcontinuationList** مع المكونات التالية:

- **targetObject** إلى اسم مميز أولي لـــ **e** DSE (قد يتضمن قيم مميزة بديلة في RDNs)؛
- **aliasedRDNs** غائبة؛
- **operationProgress** مع **nameResolutionPhase** مضبوطة على **completed** و**nextRDNtoBeResolved** غائب؛

- **rdnsResolved** غائب؛
- **referenceType** مضبوط على **subr**؛
- **accessPoints** مضبوطة على access point information المحتوية في نعت **specificKnowledge** لـــ **e'** DSE.

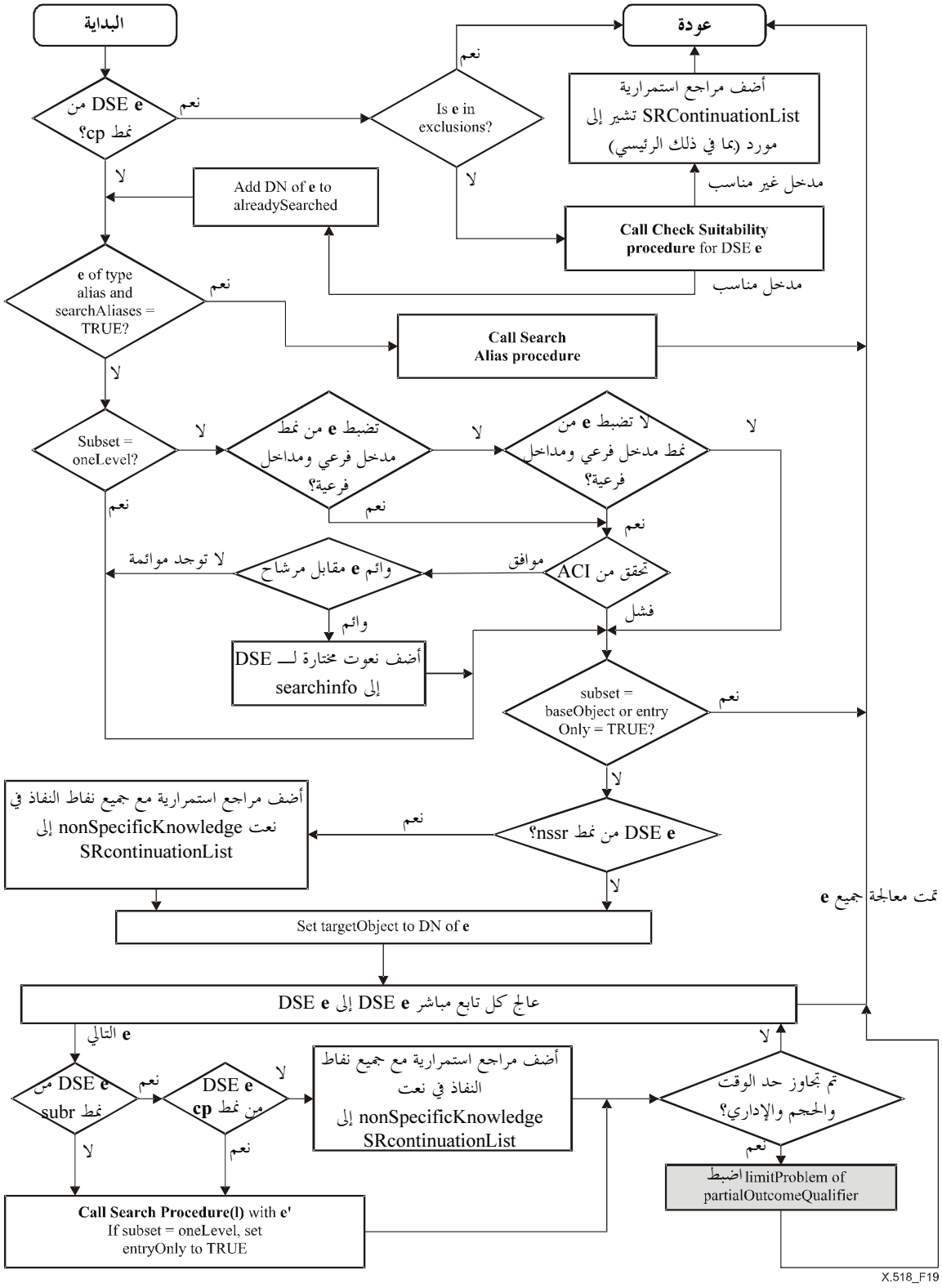
الملاحظة 5 – إذا كان **e'** هو نمط كل من **cp** و**subr**، يمكن توليد طلب فرعي لبحث محتمل من مرجع تابع أو معرفة مورد، ولكن ليس كلاهما. ويستخدم هذا الإجراء الأخير (مراجع مورد موجودة في cp).

ب) في كل الحالات:

'1' إذا كان **subset** هو **oneLevel**، اضبط **entryOnly** على **TRUE**

'2' نفذ تكرارياً إجراء **Search (I)** لـــ **e'** DSE المستهدف.

8) إذا تمت معالجة جميع التوابع، عودة إلى operation dispatcher لمزيد من المعالجة.



الشكل 19 – إجراء Search (I)

6.2.2.3.19 إجراء Search (II)

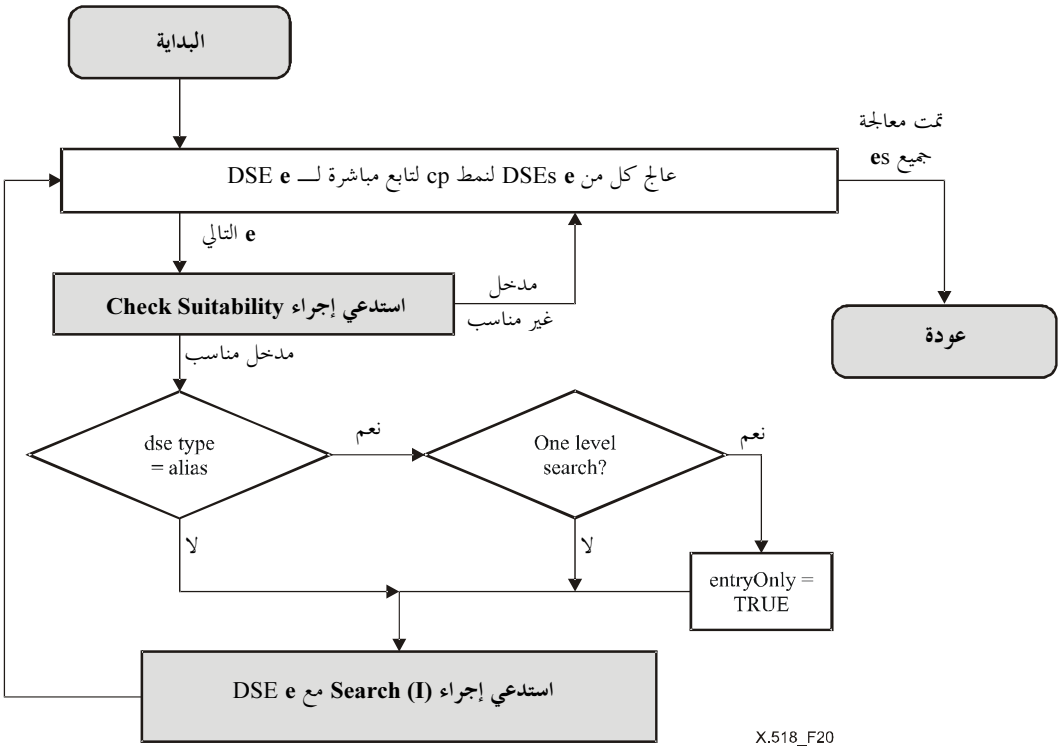
ينطبق هذا الإجراء إذا تمت معالجة طلب بحث نشأ من تجزئة طلب عند DSA استقبل منه الطلب. ويعالج الإجراء DSEs تحت DSE e' المستهدف ويستدعي إجراء **Search (I)** لكل مدخل شيء.

1) عالج جميع DSEs **e'** التي حددت مواقعها مباشرة لتابع DSE **e** المستهدف حتى تتم معالجة جميع DSEs التابعة. وإذا تمت معالجة جميع التوابع، اعد operation dispatcher لمزيد من المعالجة.

2) إذا كان DSE ليس نمط **cp**، عليك تجاهله. وعليك العودة إلى الخطوة 1.

3) استدعي **Check Suitability**. وإذا كان مناسباً، اذهب إلى الخطوة 4؛ وإلا، عليك تجاهله. والعودة إلى الخطوة 1.

4) نفذ **Search Procedure (I)** لـــ DSE **e'** كما ورد في 2.2.3.19. وإذا كان DSE من نمط **alias** وقيمة معلمة **subset** هي **oneLevel**، اضبط **ChainingArguments.entryOnly** على **TRUE** عند استدعاء إجراء **Search (I)**. وعليك العودة إلى الخطوة 1.



الشكل 20 – إجراء Search (II)

7.2.2.3.19 إجراء Search Alias

ينفذ هذا الإجراء إذا تمت مواجهة DSE لنمط **alias** خلال معالجة طلب **search** (انظر الشكل 21):

1) إذا كان **subset** هو **baseObject** أو **oneLevel**، اذهب إلى الخطوة 4.

2) إذا كان **aliasedEntryName** هو سابقة لـــ **targetObject** أو **baseObject** أو أي من القيم السابقة لـــ **targetObject** في **ChainingArguments.traceInformation**، يستثني المستعار من search لأن هذا يسبب بحثاً تكرارياً مع نتائج مستنسخة.

3) إذا كان **targetObject** أو **baseObject** أو أي من القيم السابقة لـــ **targetObject** في **ChainingArguments.traceInformation** هو سابقة لـــ **aliasedEntryName**، لا تطلب معالجة محددة للمستعار بسبب أن الشجرة الفرعية المستعارة سيجرى بحثها على أي حال.

ملاحظة – في كل من الحالتين أعلاه، قد لا يكون **baseObject** سابقة لـــ **targetObject**، نتيجة لتبديل مستعار.

4) إذا تم أداء البحث في منطقة إدارية محددة لخدمة وإذا كانت نقطة إدارية محددة لخدمة ليست سابقة لـــ **aliasedEntryName**، لا تطلب معالجة محددة للمستعار، نظرا لأن المدخل المستعار هو خارج المنطقة الإدارية المحددة لخدمة.

5) ابني طلب **DSP** مع **targetObject** مضبوط على **aliasedEntryName**. وإذا كان **subset** هو **oneLevel**، اضبط **entryOnly** على **TRUE**. واستدعي **Operation Dispatcher** للطلب لكي يستمر محلياً.

6) إذا أعاد **Operation Dispatcher** خطأ **referral** أو مشغول أو أخطاء غير متاحة، أضف (قدم واضف) مرجع استمرارية إلى **partialOutcomeQualifier** لـــ **SearchResult**، واعد.

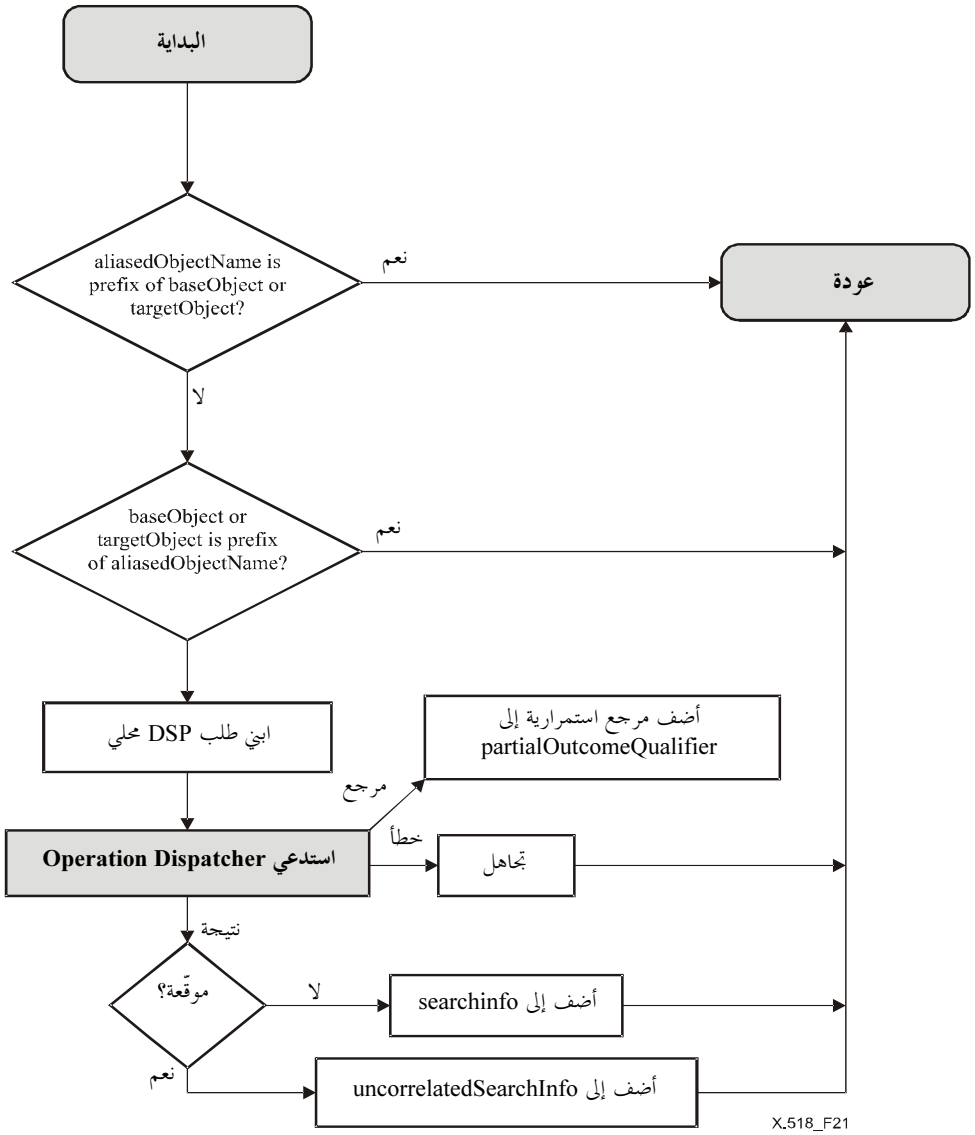
7) إذا أعاد **Operation Dispatcher** أخطاء أخرى، تجاهله واعد.

8) إذا أعاد **Operation Dispatcher** **SearchResult**:

'1' إذا كانت النتيجة موقعة أو مجفرة أو موقعة ومجفرة، أضف **uncorrelatedSearchInfo** في **SearchResult**

'2' إذا لم تكن النتيجة موقعة أو مجفرة أو موقعة ومجفرة، أضف إليها **searchInfo** في **SearchResult**.

وعد.



الشكل 21 – إجراء Search Alias

HierarchySelection procedure (I) 8.2.2.3.19

ينفذ هذا الإجراء إذا كان عضو لزمرة تراتبية تمت مواجهته خلال معالجة طلب بحث يحدد اختيار تراتبي.

أ) إذا كان اختيار تراتبي لا يدعم من قبل DSA محين، عودة مع:

- **serviceError** مع مشكلة **requestedServiceNotAvailable**؛
- نعت تبليغ **searchServiceProblem** مع القيمة **id-pr-unavailableHierarchySelect**؛
- نعت تبليغ **serviceType** له قيمة مكون **serviceType** لـــ search-rule؛
- نعت تبليغ **hierarchySelectList** يدل على اختيار (اختيارات) غير صالح.

ب) وإلا، أضف جميع المداخل المعرفة بواسطة اختيار تراتبي كما ورد في 4.2.2.3.19. وإذا نتج عن ذلك عدم إضافة مدخل، أي، تحدد الاختيارات التراتبية فقط عدم وجود مداخل، اضبط المتغير الشامل **emptyHierarchySelect**.

20 إجراءات مرجع استمرارية

تستدعي الإجراءات في هذا القسم لمعالجة قائمة مراجع استمرارية **NRcontinuationList** أو **SRcontinuationList**) التي أنشأتها إجراءات أخرى.

تتألف إجراءات **Continuation Reference** من الخطوات الواردة في الأشكال 24 و25 و26. والمرحلة الأولى هي تحديد مجموعات من مراجع استمرارية من قائمة استمرارية لها مكون شيء مستهدف مشترك. وتم إنشاء هذه من مجموعة مراجع تابعة أو تابعة غير محددة متصاحبة مع نفس المدخل في DIT. وفي كل من هذه المجموعات قد تكون هناك مراجع استمرارية تحدث أكثر من مرة واحدة. وينبغي مسح المجموعات وتجاهل أي استنساخات قد وجدت.

ويمكن معالجة هذه المجموعات (لكل منها مكون **targetObject** مختلف) على نحو مستقل، سواء تتابعياً أو بالتوازي بواسطة DSA، نظراً لعدم وجود خطر بأن تعاد نفس النتائج من أي مجموعتين. ومع ذلك، فإن معالجة كل مرجع استمرارية في مجموعة واحدة، وكل **AccessPointInformation** في مرجع استمرارية واحد، وكل نقطة نفاذ في **AccessPointInformation** واحدة، يتعين مراقبته، وإلا تحدث نتائج مستنسخة، كما يرد في 1.20.

إن الإجراء المعتمد في إجراء **APInfo** هو لمعالجة كل نقطة على حدة لمجموعة نقاط نفاذ واردة في **AccessPointInformation** واحدة. وتشير كل هذه إلى (نسخ) لنفس سياق تسمية (أو من الممكن إلى مجموعة سياقات تسمية يحتفظ بها DSA وفي حالة NSSRs). وإذا أدت نقطة النفاذ الأولى إلى نتيجة أو خطأ مفروض، ليست هناك حاجة لمزيد من نقاط نفاذ لمعالجتها. ومع ذلك، إذا كان الخطأ هو خطأ مبرمج، أي، **serviceError** (مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform** أو **invalidReference** أو **administrativeLimitExceeded**)، قد يختار DSA، كخيار محلي، معالجة نقطة نفاذ أخري من المجموعة.

إن معالجة قيم **AccessPointInformation** في مجموعة واحدة لمراجع استمرارية يتم مناولتها بطريقة موحدة، بغض النظر عن أي مراجع استمرارية نشأ منها. (هذا بسبب أن DSEs اثنين من نمط **subr** تحت مدخل وحيد ينتج مرجعين استمراريين، يحتوى كل واحد على قيمة **AccessPointInformation** واحدة، بينما DSE واحد من نمط **nssr** لنفس التابعين (بافتراض أنهما يحتفظان بهما في DSAs مختلفين) ينتج مرجع استمرارية واحد يحتوى على مجموعة من قيمتين لـــ **AccessPointInformation**).

يمكن معالجة قيم **AccessPointInformation** سواء تتابعياً أو بالتوازي، كما ورد في 1.20. ومن المحتمل أن تنتج الاستراتيجية المتوازية نتائج مستنسخة. ويجرى تجاهل الاستنساخات دائماً.

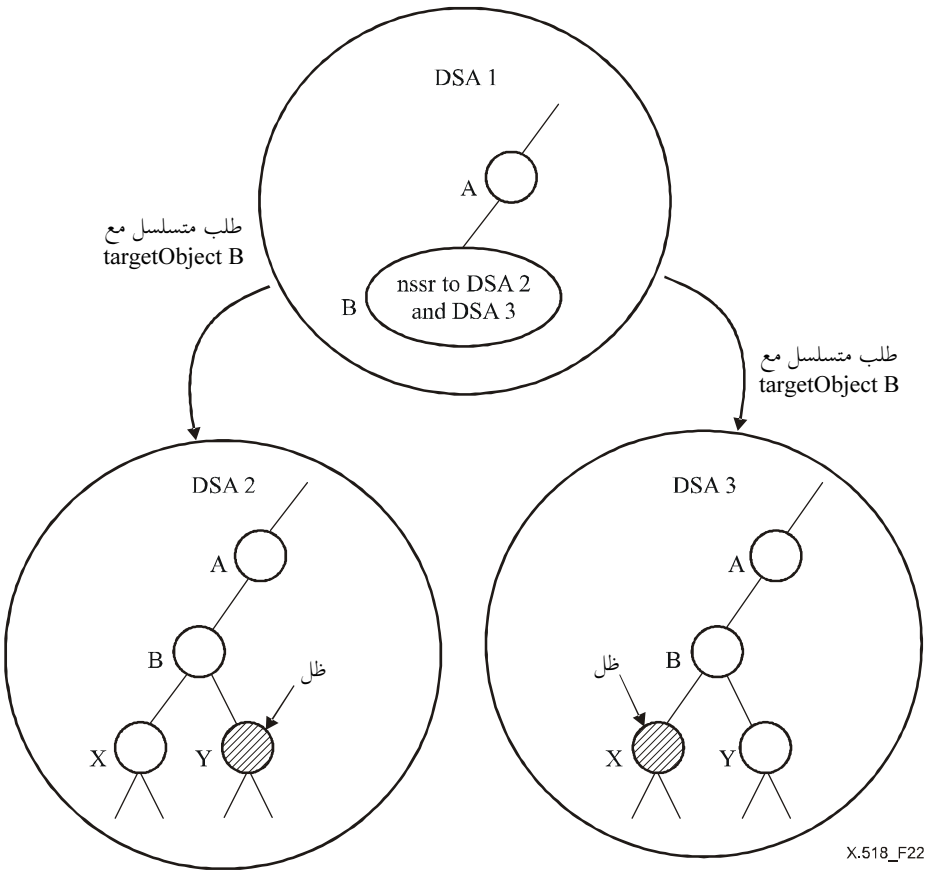
1.20 استراتيجية سلسلة في وجود تظليل

في وجود تظليل، قد يختار DSA بين استراتيجيات مختلفة عندما يكون لديه طلب سلسلة متعددة لأكثر من DSA واحد. ويحدث هذا الاختيار دائماً إذا كان على DSA أن يعالج أكثر من مرجع استمرارية واحد مع نفس **targetObject**. ويمكن أن تحدث هذه الحالة من سلسلة متعددة بواسطة تجزئة NSSR خلال Name Resolution (كما ورد في الشكل 22) أو من تجزئة طلب خلال تقييم عملية شيء متعدد (انظر الشكل 23).

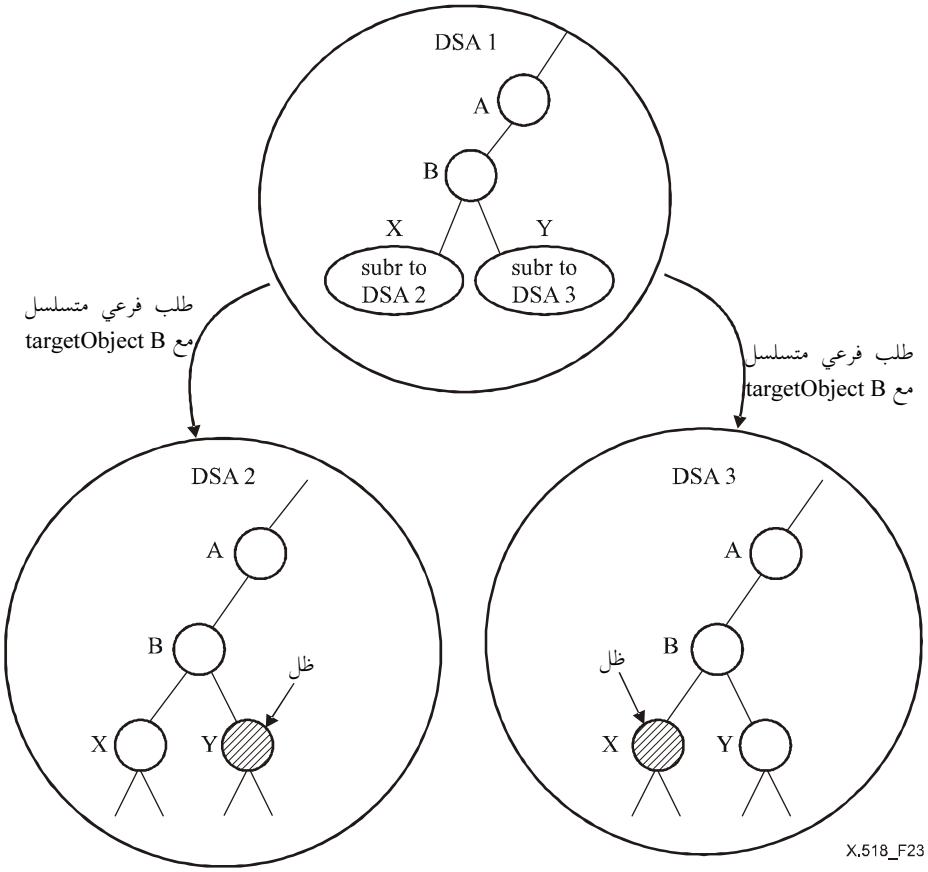
إن هدف هذه الاستراتيجيات هو تناول مشكلة النتائج المستنسخة ومعالجة الاستنساخ عندما تستخدم معلومات مظللة في سلسلة متعددة لطلبات (تسبب فيها إما NSSR أو تجزئة طلب). فمثلاً، في الشكل 22، DSA 1 يسلسل تعددياً طلب لكل من DSA 2 و3 بسبب NSSR المحتفظ به في DSE B. وإذا سمح باستخدام معلومات مظللة، يمكن لكل من DSA 2 و3 تطبيق العملية المتسلسلة على كل من الشجرتين الفرعيتين ابتداء من X و Y.

وبالمثل، في الشكل 23، DSA 1 يسلسل تعددياً (نتيجة لتجزئة طلب) المرجعين التابعين المحتفظ بهما في X DSEs و Y. ومرة ثانية، إذا سمح باستخدام معلومات مظللة، يمكن لكل من DSA 2 و3 تطبيق العملية المتسلسلة على كل من الشجرتين الفرعيتين ابتداء من X و Y.

ولتناول مشكلة الاستنساخ هذه، قد يختار DSA واحدة من الاستراتيجيات التالية عندما يسلسل تعددياً طلبات DSA متعددة مع نفس **targetObject**.



الشكل 22 – سلسلة متعددة تسبب فيها NSSR خلال Name Resolution



الشكل 23 – تجزئة طلب سلسلة متعددة باستخدام مراجع تابعة

1.1.20 استراتيجية Master only

قد يختار DSA هذه الاستراتيجية لمنع استخدام معلومات مظللة عند أداء سلسلة متعددة تتابعية أو موازية تسببت فيها تجزئة NSSR، أو تجزئة طلب خلال تقييم Search أو List. ومن أجل هذه الاستراتيجية، خلال تقييم عملية Search أو List لمكون **excludeShadows** لـــ **ChainingArguments** تضبط على **TRUE**. وإذا تمت مواجهة NSSRs خلال Name Resolution، قد يضبط DSA **nameResolveOnMaster** على **TRUE** ليضمن اتباع مسير وحيد فقط. ويضبط **nameResolveOnMaster** على **TRUE** إذا تمت مواجهة NSSR وكانت العملية هي واحدة من عمليات تعديل الدليل. وفي كلا الحالتين، يقوم DSA (s) فقط الذي يحتفظ بالمدخل (أو المداخل) الرئيسي (الأولى) المتعلق بالعملية بأداء العملية. ويمكن استخدام هذه الاستراتيجية خلال كل من سلسلة متعددة متوازية ومتتابعة.

ملاحظة – إن ضبط **nameResolveOnMaster** على **TRUE** يقضي على إمكانية مسيرات متعددة خلال استبانة اسم بواسطة:

1) تجاهل مداخل ظل ونسخ يمكن كتابتها لمداخل؛

2) بواسطة ضمان أن DSA واحد فقط يمكنه استمرار استبانة اسم في حالات حيث توزيع DIT معقدة يمكن السماح بأكثر من واحد بالاستمرار.

ويتحقق هذا بالسماح فقط لـــ DSA المحتفظ بمدخل رئيسي (أولي) متوافق مع أول RDNs **nextRDNToBeResolved** لاسم شيء مستهدف باستمرار استبانة اسم. ولن يتمكن أي DSAs من الاستمرار حتى لو كانوا يحتفظون بمداخل رئيسية تتواءم مع أكثر من اسم شيء مستهدف.

2.1.20 استراتيجية متوازية

عند استخدام هذه الاستراتيجية، يرسل DSA جميع الطلبات المتسلسلة بواسطة سلسلة متعددة متوازية. ويمكن استخدام هذه الاستراتيجية خلال تقييم Search أو List واستبانة اسم لـــ NSSRs. ويسمح هذا باستخدام معلومات مظللة لمعالجة طلبات متسلسلة، ولكن قد تنتج في تنفيذ مستنسخ ونتائج مستنسخة للعملية. وإذا اختار DSA هذه الاستراتيجية، يسحب النتائج المستنسخة من نتيجة العملية التي يعيدها.

وبسبب أن من غير الممكن سحب نتائج مستنسخة إذا طلب توقيع نتيجة، لا يختار DSA هذه الاستراتيجية إذا طلب توقيع النتائج خلال تقييم Search، ما لم يضبط **excludeShadows** أيضاً.

3.1.20 استراتيجية تتابعية

تتجنب هذه الاستراتيجية النتائج المستنسخة باستخدام سلسلة متعددة تتابعية لمعالجة طلبات (طلبات فرعية) متسلسلة لتجزئة Search أو لتجزئة NSSR. ويعالج كل طلب متسلسل الواحد بعد الآخر.

وفي حالة تجزئة NSSR، إذا اعيدت نتيجة أو خطأ مفروض، ليست هناك حاجة لتسلسل مزيد من الطلبات. وإذا أعيد خطأ مبرمج، يمكن تسلسل طلب اضافي، أو يعاد الطلب المبرمج إلى الطالب، يعتمد ذلك على السياسة المحلية.

وفي حالة تقييم Search، يضبط مكون **exclusions** لـــ **ChainingArguments** على مجموعة RDNs التي تمت معالجتها. ويتم هذا بواسطة تضمين عناصر في **ChainingResults.alreadySearched** إلى متغير **exclusions** للطلب المتسلسل التالي. وهذه هي الاستراتيجية الوحيدة التي تتجنب بالكامل الاستنساخ خلال تقييم Search.

إن الاستراتيجية التتابعية لا تعرف لتقييم List (بالرغم من امكانية استخدام سلسلة متعددة تتابعية)، نظراً لأن DSA الأعلى لا يمكنه استثناء توابع محددة من إعادتها في طلبات فرعية List (لاحظ أن **excludeShadows** لا يستثني توابع محددة، ولكن هو بالأحرى طريقة صعبة لاستثناء جميع الظلال والنسخ التي يمكن كتابتها).

2.20 اصدار طلبات فرعية متسلسلة إلى DSA عن بعد

قبل إصدار طلب فرعي، يتعين على DSA أن ينفذ عملية **dSABind** عندما يتعين على DSA أن ينشئ تصاحب لـــ DSA عن بعد. وإدارة تصاحب هي خارج مدى مواصفات الدليل. ويعتبر تصاحب DSA آخر غير متاح إذا لم يكن من الممكن إنشاء تصاحب أو يقرر DSA لأسباب محلية ألا ينشئ واحداً. وفي هذه الحالة، يكون **dSABind** قد فشل. وهو قرار محلي للتوقف عن محاولة إنشاء تصاحب وإعلان أن التصاحب غير متاح.

عندما يحاول DSA **dSABind** إلى DSA آخر، ويستقبل **directoryBindError**، يفشل إصدار الطلب الفرعي.

3.20 معلمات إجراءات

1.3.20 متغيرات

تستفيد الإجراءات من المتغيرات التالية:

- قائمة مراجع استمرارية لمعالجة **NRcontinuationList** (لإجراء **Name Resolution Continuation Reference**) و**SRcontinuationList** (لإجراءات **List Continuation Reference** و**Search Continuation Reference**، على التوالي)؛
- **CommonArguments** لمتغير العملية؛
- **ChainingArguments**.

2.3.20 نتائج

تخلق هذه الإجراءات النتائج التالية:

- قائمة بالنتائج/الأخطاء المستقبلة لطلبات متسلسلة صادرة إذا أختيرت السلسلة؛
- قائمة محينة لمراجع استمرارية غير معالجة في **continuationList**.

3.3.20 أخطاء

يمكن لهذه الإجراءات أن تعيد واحد من الأخطاء التالية:

- **serviceError** مع مشكلة **outOfScope** في حالة أن مرجع قد تم انشاؤه ليس في **scopeOfReferral**؛
- **serviceError** مع مشكلة **ditError** في حالة مرجع معرفة غير صالح تم اكتشافه؛
- **nameError** مع مشكلة **noSuchObject** في حالة أن جميع الطلبات الفرعية من تجزئة NSSR أعادت **unableToProceed**؛
- أي خطأ آخر أعيد بواسطة طلب فرعي متسلسل؛
- **referral** في حالة أن السلسلة لم تختار ويضبط **operationProgress.nameResolutionPhase** على **notStarted** أو **proceeding**.

4.20 تعريف الإجراءات

إذا ضبط **operationProgress.nameResolutionPhase** على **notStarted** أو **proceeding**، يتبع الإجراء في 1.4.20 (الإجراء **Name Resolution Continuation Reference**). وتستدعي عمليات استفسار مدخل متعدد لـــ Search أو List على التوالي الإجراءات في 2.4.20 و3.4.20.

1.4.20 الإجراء Name Resolution Continuation Reference

يتألف الإجراء **Name Resolution Continuation Reference** من الخطوات الواردة في الشكل 24. والمبدأ الأساسي لهذا الإجراء هو المعالجة التتابعية لمجموعة مراجع استمرارية نشأت خلال Name Resolution. وتنفذ الخطوات التالية لكل مرجع استمرارية **C** محتوي في **NRcontinuationList** في ترتيب مختار حتى تتم معالجة جميع المراجع أو يعاد خطأ أو نتيجة. وإذا عولجت جميع المراجع، اعد **Operation Dispatcher** لتستمر مع إجراء **Result Merging** لمعالجة نتيجة أو مرجع مستقبل.

1) تحقق من أن **chainingProhibited** مضبوط. وإذا كان كذلك، لا يسمح لـــ DSA بالتسلسل. وطبقاً لـــ **local policy**، يعاد إما **serviceError** مع مشكلة **chainingRequired** أو **referral** إلى **Operation Dispatcher**.

2) وإذا لم يضبط **chainingProhibited**، تحقق إذا كانت **local policy** تسمح بسلسلة. وإذا لم يسمح بسلسلة، عودة إلى **referral**. وإذا سمحت **local policy** بسلسلة، استمر مع الخطة التالية.

3) عالج كل من Continuation References لقائمة Continuation References التي وجدت في **NRcontinuationList**. وإذا لم توجد أي Continuation References غير معالجة، عودة مع **serviceError**.

4) عالج Continuation References **C** من **NRcontinuationList**. وإذا كان NSSR، استمر عند الخطوة 5. وإذا لم يكن NSSR، استدعي إجراء **APInfo** لمعالجته. وميز بين عمليات الإعادة الممكنة لإجراء **APInfo**:

– إذا كان إجراء **APInfo** قد أعاد **null result**، استمر عند الخطوة 3 مع معالجة Continuation Reference التالي.

– إذا كان إجراء **APInfo** قد أعاد **error** أو **referral** أو **result**، عودة.

5) وفي هذه الحالة، يكون Continuation Reference من نمط NSSR ويكون لـــ DSA خيار القيام بتسلسل تتابعي أو متوازي، يعتمد على الاختيار المحلي للاستراتيجية. وإذا عولج NSSR تتابعيا، استمر عند الخطوة 6. وإذا تعين معالجته بالتوازي، يستدعي لكل **AccessPointInformation (API)** في NSSR إجراء **APInfo** ليعالج بالتوازي. انتظر جميع API لتعالج، أي، انتظر جميع النداءات لإجراء **APInfo** للإعادة. وتحقق من أن جميع النتائج المستقبلة من نداء إجراء **APInfo** بالترتيب التالي:

– إذا أعادت جميع النداءات **serviceError** مع مشكلة **unableToProceed** ويكون **partialNameResolution** هو **FALSE**، عودة إلى **nameError**.

– إذا أعادت جميع النداءات **serviceError** مع مشكلة **unableToProceed** ويكون **partialNameResolution** هو **TRUE**، اضبط في النتيجة **partialName** على **TRUE** و**nameResolutionPhase** على **completed**، واضبط **entry suitable** (يكون هذا لـــ **lastEntryFound**) واذهب إلى التقييم الملائم للعملية.

– إذا استقبلت **results** واحدة أو أكثر، قم بـــ **discard possible duplicates** عودة إلى **result**.

– إذا استقبل **error** الذي ليس **serviceError** (مثلاً، **nameError**)، عودة إلى **error**.

– وإلا، عودة إلى **referral** أو **serviceError** إلى **Operation Dispatcher**، طبقاً للاختيار المحلي.

6) اختار **API** التالي غير المعالج من مجموعة **API** في NSSR واستمر عند الخطوة 7. وإذا تمت معالجة جميع **API**، تحقق إذا كانت جميع النداءات إلى إجراء **APInfo** اعادت **serviceError** مع مشكلة **unableToProceed**.

– إذا كانت كذلك **partialNameResolution** هو **FALSE**، ولا يمكن ايجاد المدخل ويعاد **nameError**. وإذا كانت كذلك وكان **partialnameResolution** هو **TRUE**، اضبط في النتيجة **partialName** على **TRUE** و**nameResolutionPhase** على **completed**، واضبط **entry suitable** (يكون هذا لـــ **lastEntryFound**) واذهب إلى التقييم الملائم للعملية. وإذا لم تكن كذلك، طبقاً للاختيار المحلي، اعد **referral** أو **serviceError**.

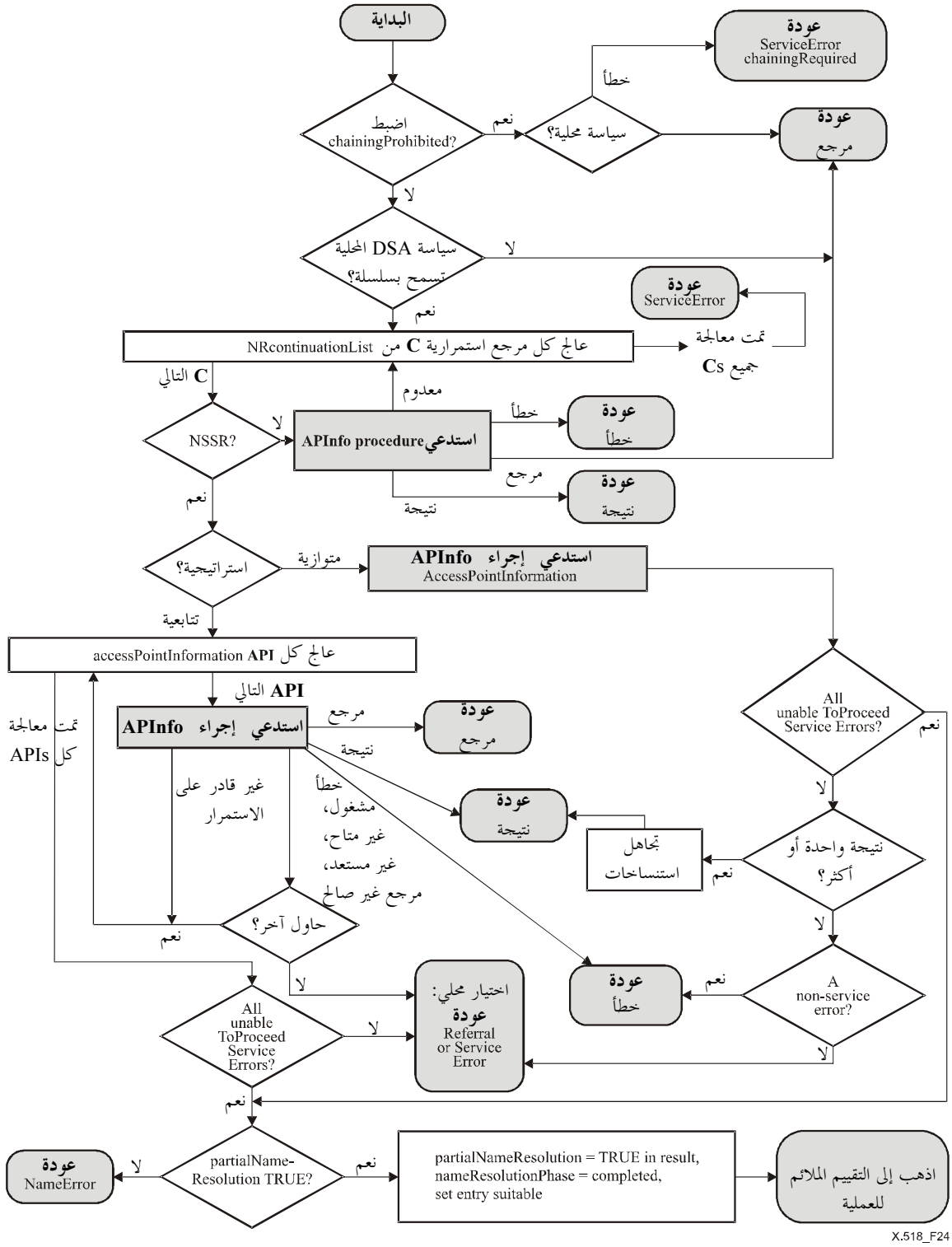
7) استدعي إجراء **APInfo**. وميز بين النتائج الممكنة من نداء إجراء **APInfo**:

– إذا استقبل **serviceError** مع مشكلة **unableToProceed**، حاول نقطة نفاذ أخرى. واستمر عند الخطوة 6.

– إذا استقبل **serviceError** مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform** أو **invalidReference**، قد تكون المشكلة ذات طابع انتقالي ويتعين على الاختيار المحلي أن يحاول ويسلسل الطلب على DSA آخر. وإذا اختير محاولة DSA آخر، استمر عند الخطوة 6؛ وإلا عودة إلى **referral** أو **serviceError**، طبقاً للاختيار المحلي.

– إذا استقبل خطأ من غير **serviceError** مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform** أو **invalidReference** أو **unableToProceed**، ينبغي إعادة الخطأ إلى **Operation Dispatcher**. وإذا كان **serviceError** هو **invalidReference**، يحول هذا إلى **ditError** قبل إعادته إلى الطالب.

– إذا استقبلت **result** أو **referral**، عودة إلى **Operation Dispatcher**.

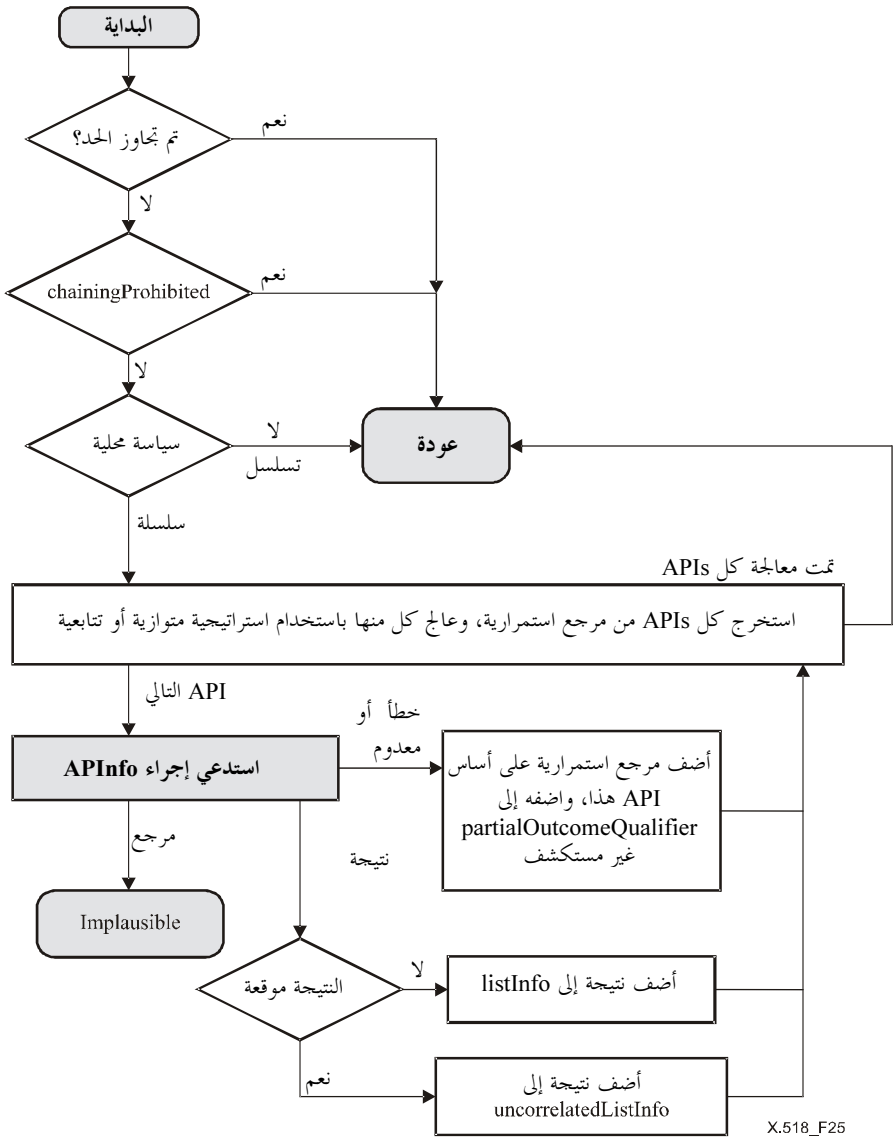


الشكل 24 – إجراء Name Resolution Continuation Reference

2.4.20 إجراء List Continuation Reference

يتألف إجراء **List Continuation Reference** من الخطوات المبينة في الشكل 25. وينفذ هذا الإجراء عندما لا يلبي طلب List في DSA المحلي وأضيفت مجموعة من مراجع استمرارية إلى **SRcontinuationList** لسلسلة أو مرجع. ويكون لجميع مراجع الاستمرارية (CR) هذه نفس **targetObject**. ويكون لـــ CRs هذه مع **referenceType nssr** قيمة (**APIs**) واحدة أو أكثر

لـــ **AccessPointInformation**، بينما يكون لـــ CRs لنمط آخر **API** واحد فقط فيهم. ويستخرج كل من **API** ويجرى اعتباره من أجل سلسلة أو مرجع.



الشكل 25 – إجراء List Continuation Reference

تنفذ الخطوات التالية:

1) إذا تم تجاوز أي مشكلة حد حتى الآن، اعد إلى **Operation Dispatcher** للاستمرار مع إجراء **Result Merging**.

2) إذا ضبط علم **chainingProhibited** في **CommonArguments.serviceControls** أو قرر DSA عدم القيام بأي تسلسل بسبب سياسته التشغيلية المحلية، يعود DSA مباشرة إلى **Operation Dispatcher** للاستمرار مع إجراء **Result Merging**.

3) أوجد مجموعة قيم لـــ **AccessPointInformation** من مكون **accessPoints** لكل مرجع استمرارية في **SRcontinuationList**.

استخدم إما استراتيجية متوازية أو تتابعية لمعالجة كل **API** كما يلي:

'1' استدعي إجراء **APInfo** مع **API** التالي في المجموعة.

'2' إذا أعيدت نتيجة أضافها إلى **listInfo** إذا لم تكن موقعة، أو أضافها إلى **uncorrelatedListInfo** إذا كانت موقعة.

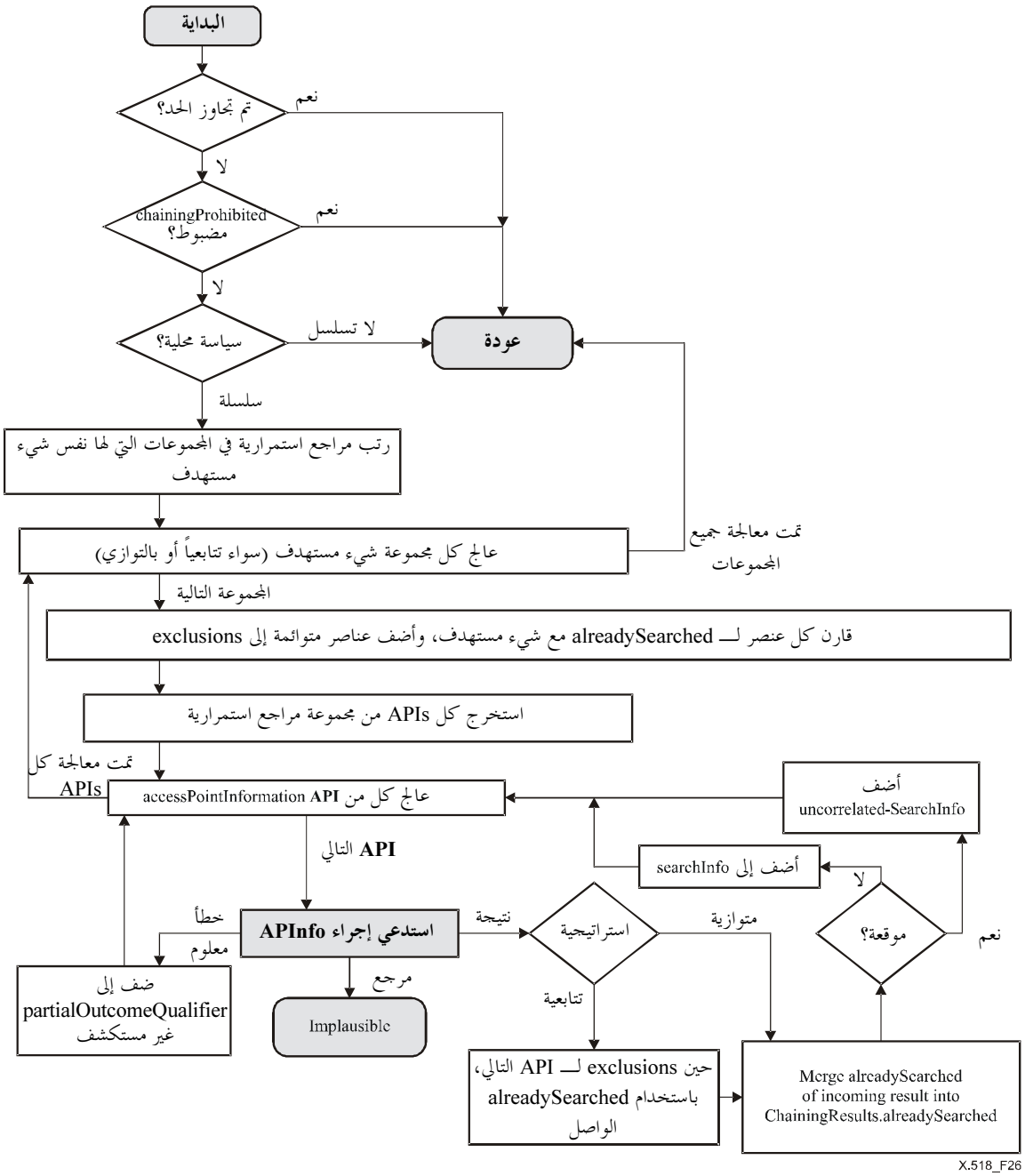
'3' إذا كانت الإعادة خطأ أو معدوم، يعني أن **APInfo** حاول جميع نقاط النفاذ في **API** دون نجاح. وعلى أساس السياسة التشغيلية المحلية وسياسة الأمن، إما أن تتجاهل وتستمر إلى **API** التالي، أو أضف مرجع استمرارية على أساس **API** هذا إلى **partialOutcomeQualifier**.

ملاحظة – ليس من المستحسن استعادة مرجع من **APInfo**. وينبغي أن يأتي أي "مرجع" على شكل **unexplored** في **partialOutcomeQualifier**.

4) وعند معالجة جميع APIs، عودة إلى **Operation Dispatcher**.

3.4.20 إجراء Search Continuation Reference

يتألف إجراء **Search Continuation Reference** من الخطوات المبينة في الشكل 26. وينفذ هذا الإجراء عندما لا يلبي طلب Search في DSA المحلي وأضيفت مجموعة من مراجع استمرارية إلى **SRcontinuationList** لسلسلة أو مرجع. والإجراء مماثل لإجراء **List Continuation Reference**. والفرق هو أن في هذه الحالة قد يكون لمراجع استمرارية في **SRcontinuationList** قيم **targetObject** مختلفة. ومن ثم، تخزن مراجع استمرارية في مجموعات مراجع استمرارية مع نفس **targetObject**. وأيضاً، يعرف استخدام **exclusions** في متغيرات سلسلة و**alreadySearched** في نتائج سلسلة، نظراً لأن هذه استراتيجية مهمة للبحث. إن استخدام **exclusions** و**alreadySearched** يطبق لمعالجة كل مجموعة مراجع استمرارية مع نفس **targetObject**.



الشكل 26 – إجراء Search Continuation Reference

تنفذ الخطوات التالية:

1) إذا تم تجاوز أي مشكلة حد حتى الآن، عودة إلى **Operation Dispatcher** للاستمرار مع إجراء **Result Merging**.

2) إذا ضبط علم **chainingProhibited** في **CommonArguments.serviceControls** أو قرر DSA عدم القيام بأي تسلسل بسبب سياسته التشغيلية المحلية، يعود DSA مباشرة إلى **Operation Dispatcher** للاستمرار مع إجراء **Result Merging**.

3) رتب مراجع استمرارية في **SRcontinuationList** في مجموعات لها نفس **targetObject**. ولا يشمل نمط **ditBridge** المراجع الاستمرارية في هذه المجموعات، ويشكل كل مرجع استمرارية مجموعة في حد ذاتها. وفي كل مجموعة، اسحب أي استنساخات.

الملاحظة 1 – إذا كانت قيمة واحدة أو أكثر لـــ **targetObject** ليست RDN أولي، قد لا يكون هذا الترتيب دقيقاً. ويأخذ الترتيب في عين الاعتبار RDNs مميزة بديلة، إذا عرفت.

4) لكل مجموعة فرعية لمراجع استمرارية أنشئ مجموعة لقيم AccessPointInformation من مكون accessPoints لكل مرجع استمرارية في المجموعة الفرعية، واختر إما استراتيجية تتابعية أو متوازية لمزيد من المعالجة. وإذا اختيرت الاستراتيجية المتوازية، تخطى الخطوات أدناه التي تدل على قابلية تطبيق الاستراتيجية التتابعية فقط.

أ) إذا اختيرت الاستراتيجية التتبعية، حافظ على متغير محلي **localExclusions** لكل مجموعة من مراجع استمرارية لها نفس **targetObject**. وابتداء، يضبط **localExclusions** على **exclusions** لطلب السلسلة الواصل (إذا وجد)، وجميع الأشجار الفرعية التي تم بحثها محلياً تحت **targetObject** مباشرة.

ب) إذا استخدمت الاستراتيجية التتابعية، قارن **targetObject** بجميع عناصر **localExclusions**، واسحب تلك العناصر التي لا تحتوي على **targetObject** كسابقة. وهذه هي **exclusions** ذات العلاقة لشيء مستهدف حالي.

ج) استخرج جميع **APIs** من جميع مراجع استمرارية لمجموعة شيء مستهدف حالي.

د) عروة من خلال كل **API**. ولكل **API**:

'1' استدعي **APInfo**.

'2' إذا أعيدت نتيجة، أضف النتيجة إلى **searchInfo** إذا لم تكن موقعة، أو أضف إلى **uncorrelatedSearchInfo** إذا كانت موقعة. وإذا استخدمت الاستراتيجية التتابعية، حين **localExclusions** باستخدام **alreadySearched** في الإجابة الواصلة، ودمج **alreadySearched** في الإجابة الواصلة إلى **ChainingResults.alreadySearched** لـــ DSA هذا. ثم استمر مع **API** التالية.

'3' إذا أعيد خطأ أو معدوم، يعني أن **APInfo** حاولت جميع نقاط النفاذ في **API** دون نجاح. وعلى أساس السياسة التشغيلية المحلية وسياسة الأمن، إما أن تتجاهل وتستمر إلى **API** التالي، أو تضيف مرجع استمرارية على أساس **API** هذا إلى **partialOutcomeQualifier**.

ملاحظة – ليس من المستحسن استعادة مرجع من **APInfo**. وينبغي أن يأتي أي "مرجع" على شكل **unexplored** في **partialOutcomeQualifier**.

هـ) عند معالجة جميع **API**، استمر مع المجموعة التالية من مراجع استمرارية مع نفس **targetObject**.

5) وعند معالجة جميع **API**، عودة إلى **Operation Dispatcher**.

4.4.20 إجراء APInfo

يستدعي هذا الإجراء لمعالجة **AccessPointInformation** التي تحتوي على نقطة نفاذ واحدة أو أكثر (انظر الشكل 27). ويعالج كل مرجع على حدة حتى تعاد إما نتيجة أو خطأ. وإذا كان الخطأ هو خطأ خدمة بحيث يمكن أن تنجح محاولة نقطة نفاذ أخرى، تجرى محاولة نقاط نفاذ إضافية طالما تسمح سياسة تشغيلية محلية بذلك:

1) قم بأداء اكتشاف عروة. وإذا اكتشفت عروة، عودة إلى **serviceError** مع مشكلة **loopDetected**. وإلاّ، استمر عند الخطوة 2.

2) عالج كل نقطة نفاذ من معلومات نقطة نفاذ. وإذا تمت معالجة الجميع، عودة إلى **null result**. وإذا كانت هناك أي نقطة نفاذ لمعالجتها، استمر عند الخطوة 3.

3) تحقق ما إذا كانت السياسة المحلية تسمح بسلسلة لنقطة النفاذ هذه. وينبغي أن يأخذ التحقق هذا في عين الاعتبار بيئة مراقبة الخدمة ومتغيرات السلسلة (مثل، **chainingProhibited** و**preferChaining**، سواء أكانت نقطة النفاذ في **localScope** أم لا، **excludeShadows**). وإذا لم تسمح السياسة المحلية أو بيئة مراقبة الخدمة باستخدام نقطة النفاذ المعينة هذه، تجاهل نقطة النفاذ واستمر عند الخطوة 2. وإذا كان من الممكن استخدام نقطة النفاذ، استمر عند الخطوة 4.

4) إذا اختارت السياسةُ المحلية الاستراتيجيةَ الرئيسيةَ فقط، اضبط متغير سلسلة **excludeShadows** على **TRUE**.

إذا لم يكن **nameResolutionPhase** هو **completed** وهي الاستراتيجية استبانة اسم على مداخل رئيسية، اضبط **nameResolveOnMaster** على **TRUE**.

يضبط متغير السلسلة **nameResolveOnMaster** على **TRUE** إذا كان أي من التالي هو true:

- في متغير سلسلة واصل **nameResolutionPhase** هو **proceeding** و**nameResolveOnMaster** هو **TRUE**؛ أو
- العملية هي من عمليات تعديل، يكون **referenceType** لطلب سلسلة يتعين اصداره هو NSSR، تستخدم استراتيجية متوازية.

ملاحظة – إن هذه طريقة استخدام **nameResolveOnMaster** هذه هي لمنع عمليات تعديل تطبق لمرات عديدة نتيجة لوجود NSSR.

5) ابن طلب متسلسل وحاول اصداره:

أ) قم بأداء تجنب عروة بواسطة التحقق إذا كان بند له نفس **targetObject** و**operationProgress** يحث في **traceInformation** لـــ **ChainingArguments** مستقبل. وإذا كان الطلب الناتج (كما ورد في الخطوة 5) يؤدى إلى عروة، يعيد DSA إما **serviceError** مع مشكلة **loopDetected**. إلى DUA/ لزبون LDAP /DSA الطالب أو تجاهل نقطة النفاذ وحاول نقطة النفاذ التالية بالاستمرار عند الخطوة 2.

ب) إذا كان الطلب أو الطلب الفرعي الذي يتسلسل هو نتيجة تنفيذ مرجع، يُطلب تحقق اضافي لتجنب عروة. وتحقق من أن بنداً مع نفس **targetObject** و**operationProgress** وDSA مستهدف يحدث في **referralRequests**. وإذا كان كذلك، اتخذ الإجراء المحدد في أ). وإذا لم يكن، أضف **TraceItem** جديد إلى **referralRequests** مع المكونات التالية:

- يضبط **targetObject** و**operationProgress** على قيمة طلب/طلب فرعي متسلسل؛
- يضبط **dsa** على اسم DSA الذي يتسلسل طلبه/طلبه الفرعي.

ج) بعد Bind ناجح، يصدر DSA عملية متسلسلة لنفس نمط العملية باعتبارها العملية التي تعالج مع المعلمات التالية:

- يضبط متغير العملية في العملية المتسلسلة كما لمتغير العملية المستقبل؛
- يضبط **ChainingArguments.originator** كما يستقبل؛
- يضبط **ChainingArguments.targetObject** على **targetObject** لمرجع استمرارية؛

- يضبط **ChainingArguments.operationProgress** على قيمة **operationProgress** لمرجع استمرارية؛
- يضبط **ChainingArguments.traceInformation** على معلومات أثر كما حينت بواسطة إجراء **Request Validation** إذا كان مرجع الاستمرارية ليس من نمط **ditBridge**، وإلا يكون المكون غائباً؛
- **ChainingArguments.aliasDereferenced** على القيمة المحينة لـــ **aliasDereferenced** محين محلياً؛
- **ChainingArguments.returnCrossRefs** على اختيار محلي؛
- **ChainingArguments.referenceType** على قيمة **referenceType** لمرجع استمرارية؛
- **ChainingArguments.timeLimit** على قيمة **timeLimit** المستقبلة؛
- يضبط **chainingArguments.exclusions** على إما exclusions المتعلق بشيء مستهدف حالي إذا استدعي إجراء Search Continuation Reference أو غائب إذا استدعي إجراء **APInfo** من قبل Name Resolution أو إجراءات List Continuation؛
- يضبط **SecurityParameters** على قيمة **SecurityParameters** المستقبلة.

6) إذا لم يكن من الممكن إصدار الطلب بنجاح، استمر عند الخطوة 7. وإذا كان من الممكن إصداره بنجاح، استمر عند الخطوة 8.

7) يكون اختيار محلي سواء تم الاستمرار أم لا. وإذا اختار DSA أن يستمر، يتم تجاهل الخطأ وتجرى محاولة نقطة النفاذ التالية. واستمر عند النقطة 2. وإذا قرر DSA ألا يحاول نقطة نفاذ أخرى، يكون اختيار السياسة المحلية إعادة **referral** أو **serviceError** إلى طالب الإجراء.

8) وإذا كان من الممكن إصدار الطلب بنجاح، ينتظر DSA الإجابة ويعالجه:

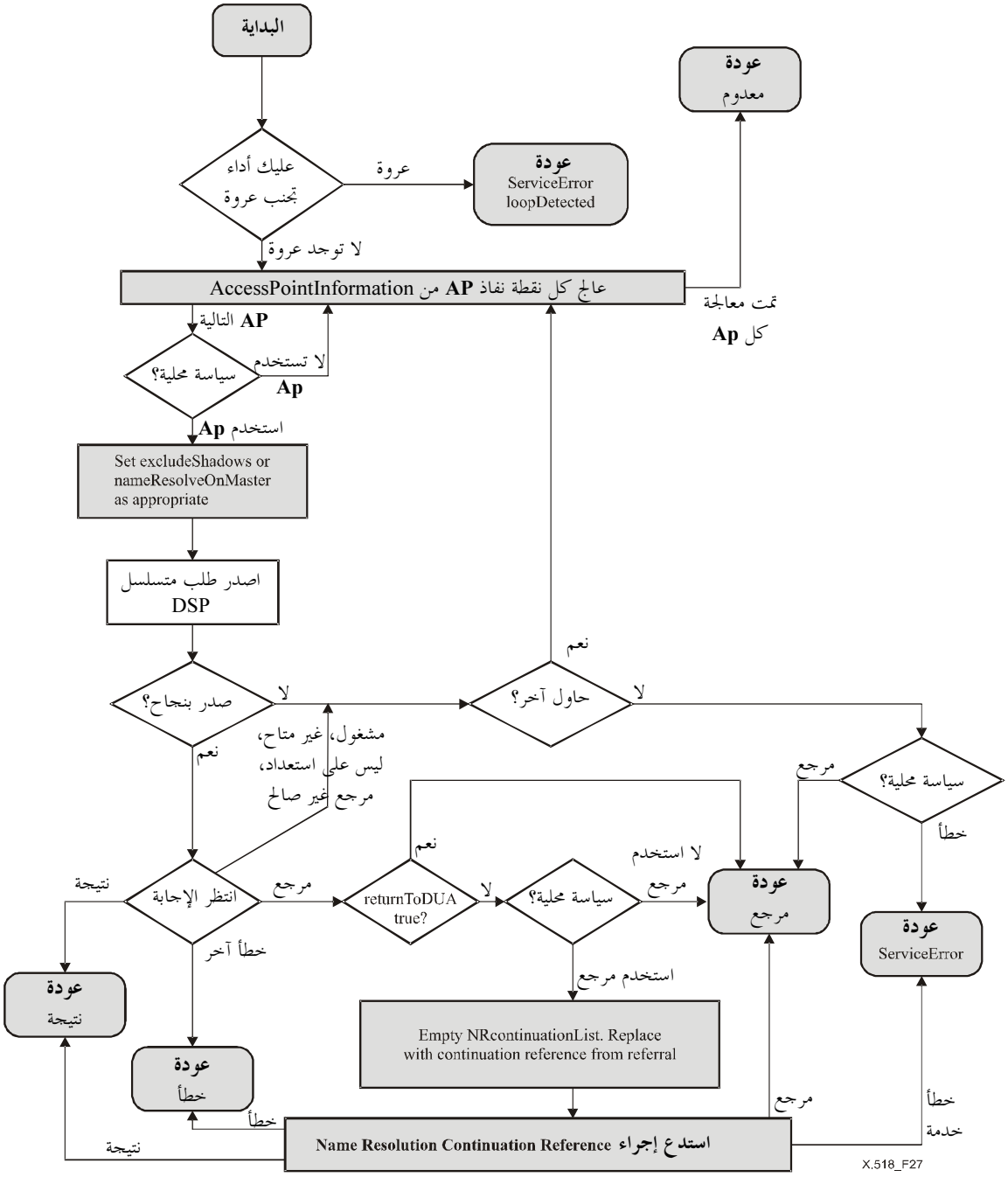
أ) إذا استقبلت **result**، تعاد **result** إلى طالب الإجراء.

ب) إذا استقبل **serviceError** مع مشكلة **busy** أو **unavailable** أو **unwillingToPerform** أو **invalidReference**، استمر عند الخطوة 7.

ج) إذا استقبل **referral** وضبط **returnToDUA** على **TRUE**، لا يعمل DSA المستقبل على Referral ولكن يعيد Referral إلى الطالب.

د) إذا استقبل **referral** وضبط **returnToDUA** على **FALSE**، تنطبق نفس اعتبارات السياسة المحلية كما في الخطوة 3 (مع الأخذ في الاعتبار مراقبة خدمة ومتغيرات سلسلة واستراتيجية سلسلة وما إلى ذلك). وإذا تقرر ألا يبدل **referral**، أعد **referral** إلى الطالب. وإذا تقرر تبديل **referral**، فرغ **NRcontinuationList**، وضع Continuation Reference كما استقبل في Referral في **NRcontinuationList** واستدعي إجراء **Name Resolution Continuation Reference**. وقد يؤدى هذا إلى **result** أو **referral** أو **serviceError** أو **error** آخر. وما يستقبل من نداء **Name Resolution Continuation Reference** يعاد إلى الطالب.

هـ) إذا حدث أي **error** آخر، يعاد إلى الطالب.



الشكل 27 – إجراء APInfo

5.20 إجراء Abandon

ينفذ هذا الإجراء إذا استقبل طلب ترك. ويتألف من الخطوات التالية كما يرد في الشكل 28:

1) عندما يستقبل طلب **abandon**، الذي يشير إلى عملية غير معروفة، يعاد **abandonFailed** مع مشكلة **noSuchOperation** إلى الطالب.

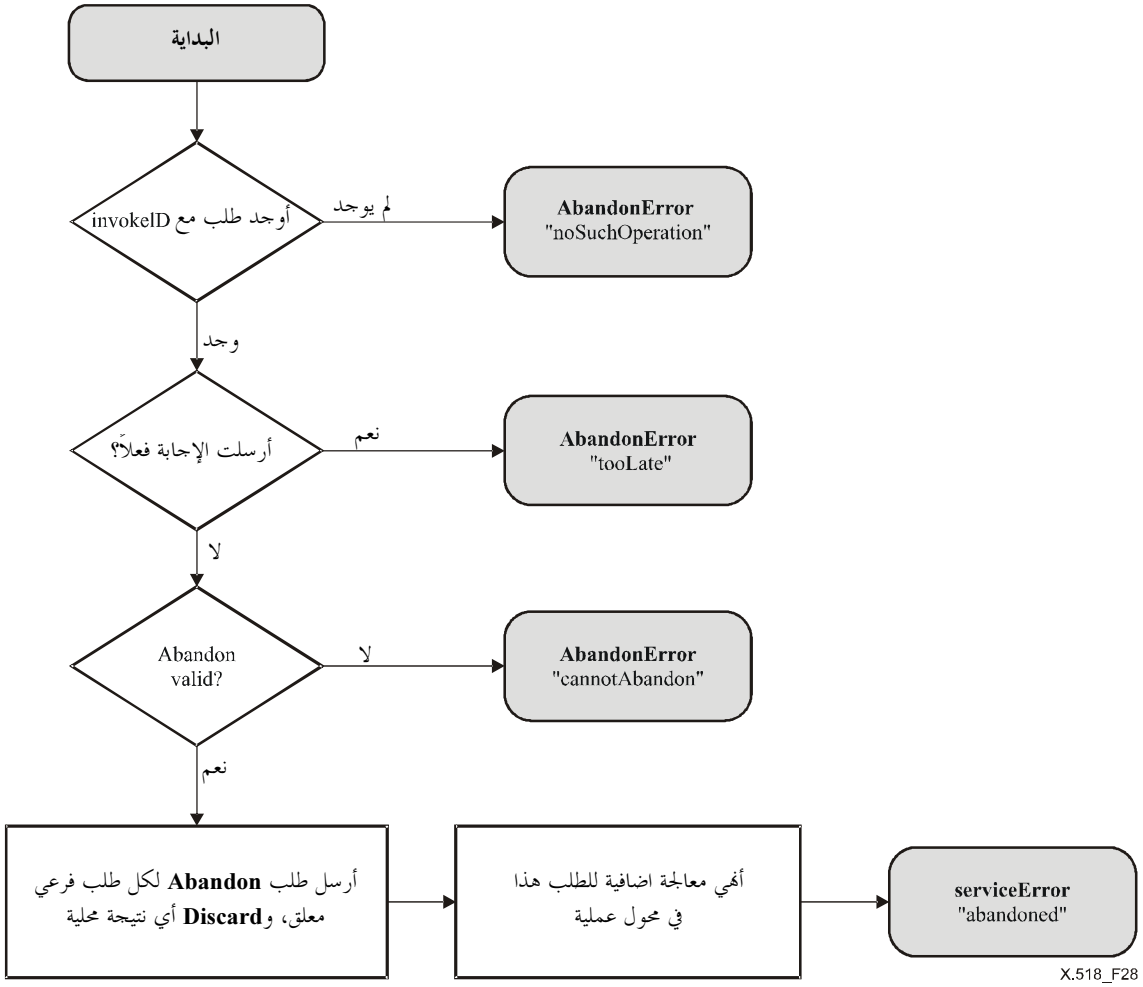
2) إذا تمت الإجابة على الطلب الذي يتعين تركه، واحتفظ DSA بمعلومات لمعرفة ذلك، يعاد **abandonError** مع مشكلة **tooLate** إلى الطالب.

3) إذا كان طلب Abandon غير صالح، أي، يطلب طلب ترك ليس طلب استفسار، يعاد **abandonFailed** مع مشكلة **cannotAbandon** إلى الطالب.

4) إذا كان لـــ DSA طلبات (طلبات فرعية) متسلسلة معلقة عندما استقبل طلب Abandon للطلب الأصلي، يقرر DSA أن يحاول الترك، ويمكن أن يرسل طلبات Abandon إلى لا أحد، أو بعض أو جميع الطلبات (الطلبات الفرعية) للعملية قيد النظر، ثم ينتظر طلب Abandon والطلبات (الطلبات الفرعية) المعلقة. وفي أي وقت خلال هذه العملية يمكن لـــ DSA أن يرسل نتيجة Abandon و**abandonFailed** إلى الطالب ثم يتجاهل الإجابات على طلبات Abandon الصادرة والطلبات (الطلبات الفرعية) المعلقة عندما تصل.

إذا قرر DSA عدم إرسال إجابات إلى الطالب حتى لم تعد هناك طلبات (طلبات فرعية) معلقة، يمكن أن يرسل اختيارياً خطأ **abandonedFailed** إلى الطالب إذا كانت جميع طلبات **abandon** الصادرة تمت الإجابة عليها مع أخطاء **abandonedFailed** و لم يتم أداء عملية ترك محلية.

إذا أعيد خطأ **AbandonedFailed** إلى الطالب، يعامل الطلب الأصلي كما لو كان طلب Abandon لم يتم استقباله أبداً.



الشكل 28 – إجراء Abandon

21 إجراء Results Merging

يستدعي إجراء **Result Merging** في الشكل 29 ليتبع واحد من إجراءات **Continuation Reference**. ويسحب هذا الاجراء الاستنساخات، إذا لم تكن النتيجة موقعة، وإذا كانت هناك مراجع استمرارية اضافية في **partialOutcomeQualifier.unexplored**. ثم يستدعي إجراء (إجراءات) Continuation Reference، إذا سمحت السياسة التشغيلية المحلية:

1) إذا كانت العملية هي عملية List، استمر عند الخطوة 2؛ وإذا كانت العملية هي عملية Search، استمر عند الخطوة 3؛ وإلا، أعد النتيجة التي تم توريدها كمعلمة دخل إلى إجراء **Result Merging**.

2) تكون العملية هي عملية List. اسحب جميع المستنسخات، مع إيلاء الأفضلية لمعلومات رئيسية على معلومات ظل.

إذا كانت نتيجة العملية مولدة محلياً وتحتوي على Continuation References، لن تستخدم هذه للتسلسل ولكن تعاد إلى المستعمل. وفي هذه الحالة، استمر عند الخطوة 6.

إذا تم استقبال نتيجة العملية كنتيجة عملية Chained List، قد تحتوي النتيجة على Continuation References. وفي هذه الحالة، تحقق إذا كانت مراقبة خدمة **preferChaining** قد ضبطت. وإذا كانت **TRUE**، ينبغي استخدام Continuation References للتسلسل بواسطة DSA . واستمر عند الخطوة 4.

3) تكون العملية هي عملية Search. اسحب جميع المستنسخات، مع إيلاء الأفضلية لمعلومات رئيسية على معلومات ظل. وإذا كانت هناك مشكلة حد أعد النتيجة. وإلا، استمر عند الخطوة 4.

4) عالج كل Continuation Reference يكون في **partialOutcomeQualifier.unexplored** لنتيجة أي عملية متسلسلة. وإذا قررت السياسة المحلية عدم استخدامه للتسلسل، تجاهله واختار Continuation Reference آخر. وإذا سمحت السياسة المحلية باستخدام Continuation Reference للتسلسل، قم بتأدية ما يلي:

تحقق من أن **nameResolutionPhase** الذي ورد في Continuation Reference. وإذا كان **notStarted** أو **proceeding**، اضفه إلى قائمة Continuation References التي ستورد إلى إجراء **Name Resolution Continuation (NRcontinuationList)**. وإذا كان **nameResolutionPhase** هو **completed** أضف Continuation Reference إلى قائمة Continuation References التي ستورد إلى إجراء Continuation لطلب فرعي **(SRcontinuationList)**.

استمر حتى تتم معالجة جميع Continuation References.

5) إذا كانت هناك Continuation References لتعالج في **SRcontinuationList**، تحقق من نمط العملية. وإذا كانت العملية هي عملية List، استدع إجراء **List Continuation Reference** واستمر عند الخطوة 2. وإذا كانت العملية هي عملية Search، استدع إجراء **Search Continuation Reference** واستمر عند الخطوة 3.

إذا كان **SRcontinuationList** فارغاً، تحقق إذا كان هناك Continuation References في **NRcontinuationList**. وإذا كان الأمر كذلك، استدع **Name Resolution Continuation Reference** واستمر عند الخطوة 3.

إذا كانت كل من قائمتي continuation فارغة، استمر عند الخطوة 6.

6) تحقق ما إذا كانت النتيجة فارغة. وإذا لم تكن فارغة، أعدها. وإذا كانت فارغة، إما أن تعيد **null result** إذا كانت مراقبة نفاذ والسياسة المحلية تسمحان بذلك، أو أعد خطأ ملائم.

في حالة استقبال DSA نتائج List أو Search من DSAs آخرين ولهذه النتائج معلمات غير معروفة لـــ DSA، تعاد النتائج غير المترابطة. وإلا، يؤدى DSA دمج، إذا لم تكن نتائج البحث موقعة، أو إذا كان DSA هو المنفذ الإبتدائي الذي يسمح له بسحب التوقيعات (انظر 9.7 من ISO/IEC 9594-3 | ITU-T X.511).

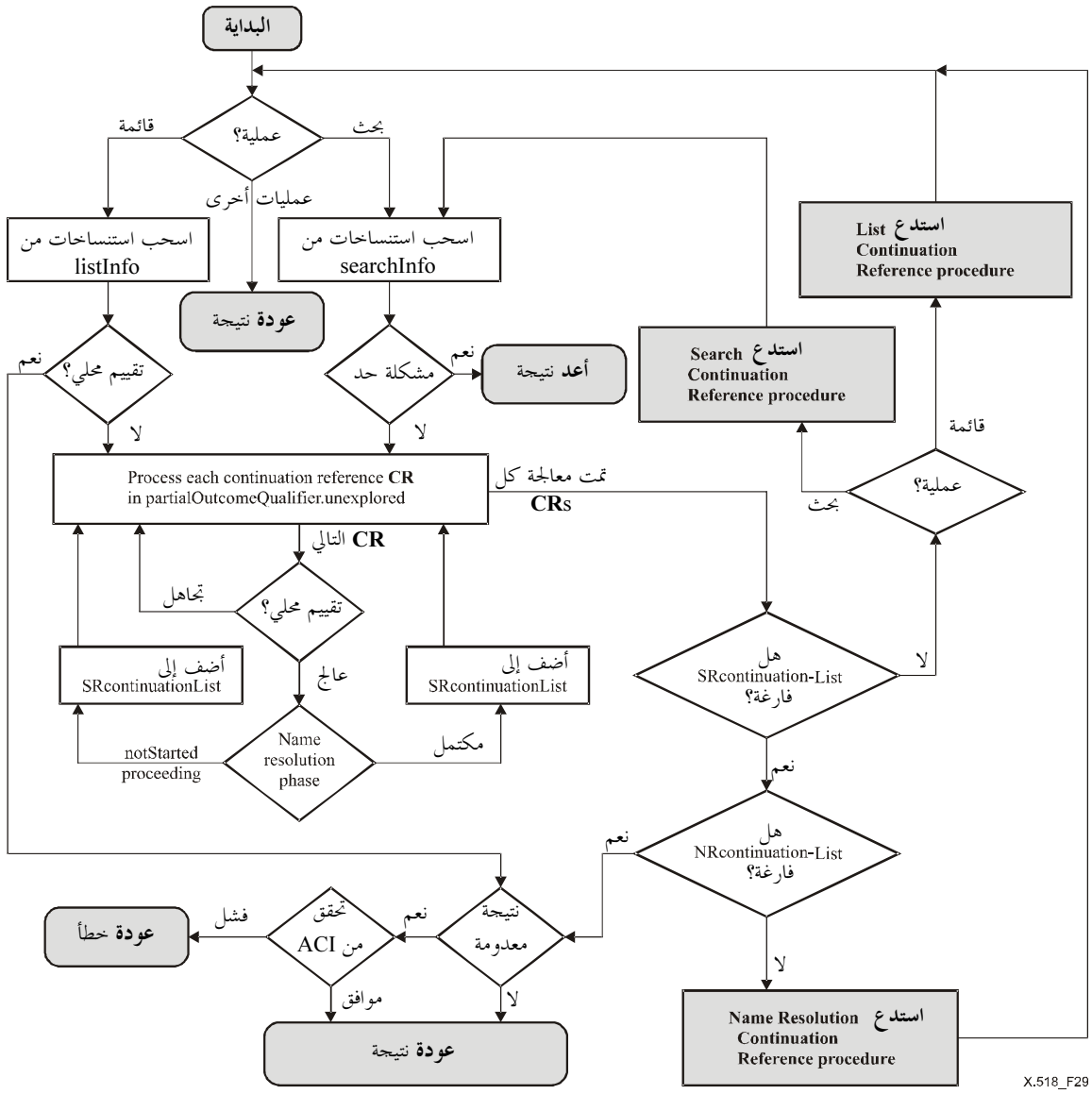
إذا استقبل DSA نتائج غير موقعة غير مترابطة من DSA غير قادر على أداء توحيد، يقوم بالدمج، إذا كان لديه المعرفة الصحيحة لجميع معلمات النتائج غير المترابطة.

إذا استقبل DSA نتائج غير موقعة من DSAs آخرين، من الممكن أن له نتيجة محلية أيضاً وعند توليد عد مدخل يتعين إعادته في **entryCount** لـــ **PartialOutcomeQualifier** ولده DSA، يأخذ DSA مجموع قيم **entryCount** المستقبلة، والنتيجة المحلية وعدد المداخل المستقبلة من DSAs التي لم تعيد قيمة **entryCount** ثم يعوض المداخل المستنسخة. وإذا كان DSA هو المنفذ الابتدائي وطلبت النتائج المتصفحة، يشمل أيضاً عد المدخل لتوقيع النتائج من DSAs آخرين.

إذا طلبت نتائج متصفحة و لم يتم مواجهة مشكلة حد من قبل أي من DSA، يأخذ DSA اختيار **exact** لمعلمة **entryCount**. وتعطي نفس القيمة لكل صفحة معادة.

إذا واجه DSA واحد أو أكثر مشكلة حد:

- إذا كان جميع DSAs الذين واجهوا مشكلة حد قد أعادوا **entryCount** مع **exact** أو اختيار **bestEstimate**، يأخذ اختيار **bestEstimate** إذا كان DSA واحد قد أخذ ذلك الاختيار. وإلا، يأخذ اختيار **exact**.
- إذا واجه DSA واحد مشكلة حد وأعاد **entryCount** مع اختيار **bestEstimate** أو لم يعيد أي **entryCount**، يأخذ اختيار **lowEstimate**.



الشكل 29 – إجراء Results Merging

22 إجراءات استيقان موزع

يحدد هذا القسم الإجراءات الضرورية لدعم خدمات استيقان موزع لدليل. وتقسم هذه الخدمات، ومن ثم الإجراءات، إلى فئات كما يلي:

- استيقان مرسل، الذي يدعم إما في شكل غير محمي (على أساس هوية بسيطة) أو شكل مضمون (على أساس توقيعات رقمية)؛
- استيقان نتائج تكون محمية بالمثل (مرة ثانية قائمة على توقيعات رقمية).

1.22 استيقان مرسل

1.1.22 استيقان على أساس الهوية

تمكن خدمة استيقان على أساس الهوية DSAs من استيقان الطالب الأصلي للمعلومات لغرض تنفيذ مراقبة نفاذ محلية. ويعتمد DSA الذي يرغب في استغلال هذه الخدمة من الإجراء التالي:

- بالنسبة لـــ DSA الطالب لاستيقان طلب DAP أو LDAP، يحصل DSA على اسم مميز للطالب من خلال إنشاء إجراءات Bind في وقت تصاحب DUA (DUA إلى DSA) أو تصاحب زبون LDAP (زبون LDAP إلى DSA). والانتهاء الناجح لهذه الإجراءات لا تمس بأي حال من الأحوال بمستوى الاستيقان الذي قد يطلب فيما بعد لمعالجة عمليات باستخدام ذلك التصاحب.
- يدرج DSA الذي يوجد مع تصاحب DUA أو زبون LDAP الاسم المميز للطالب في المجال الابتدائي لـــ **ChainingArguments** لجميع العمليات المتسلسلة التالية بـــ DSAs الآخرين.
- قد يلبي DSA، عند استقباله لعملية متسلسلة، تلك العملية، أولاً، يعتمد على تحديد حقوق النفاذ (آلية معرفة محلياً). وإذا كان الناتج غير مرضي، يمكن إعادة **securityError** مع مشكلة **insufficientAccessRights**.

2.1.22 استيقان مرسل على أساس التوقيع

تمكن خدمة استيقان على أساس التوقيع DSA من استيقان (بطريقة آمنة) مرسل طلب خدمة معينة. والإجراءات التي ينفذها DSA لتحقيق هذه الخدمة ترد في هذا القسم.

ينفذ خدمة استيقان على أساس التوقيع DUA باستخدام متغير **PROTECTED** لطلب خدمة محمية اختيارياً مع **DirQOP signed** أو **signedAndEncrypted**.

إن DSA، عند استقباله لطلب موقع من DSA آخر، يشطب توقيع DSA قبل معالجة العملية. ومع افتراض أن التحقق من أي توقيع يثبت أنه مرضي، يستمر DSA في التقدم بالعملية. وإذا احتاج DSA، خلال المعالجة، إلى أداء تسلسل، تبني مجموعة متغير لكل عملية متسلسلة متصاحبة كما يلي:

- يشكل DSA مجموعة متغير قد يكون موقعاً؛ وتتألف مجموعة المتغير من مجموعة متغير موقع واصل مع **ChainingArguments** معدلة.

وفي حالة قدرة DSA على المساهمة بمعلومات في الاستجابة، يمكن استخدام استيقان مرسل، على أساس طلب خدمة موقعة، لتحديد حقوق النفاذ لتلك المعلومات.

إذا استقبل DSA طلب خدمة غير موقع لمعلومات يمكن الافصاح عنها فقط بناءً على استيقان مرسل، يعاد **securityError** مع مشكلة **protectionRequired**.

2.22 استيقان نتائج

تتوفر هذه الخدمة لتمكين الطالبين لعمليات دليل (DUAs أو زبون LDAP أو DSAs) التحقق (بطريقة آمنة باستخدام تقنيات توقيع رقمي) من مصدر نتائج. وقد تطلب خدمة استيقان نتائج بغض النظر عما إذا كان استيقان مرسل سيستخدم.

تبدأ خدمة استيقان نتائج باستخدام القيمة الموقعة لمكون **protectionRequest** كما وردت في مجموعة المتغير لعمليات الدليل؛ ويمكن لـــ DSA مستقبل لعملية مع هذا الخيار المختار أن يوقع اختيارياً أي نتائج تالية. ويقوم الخيار الموقع لطلب الحماية بدور دلالة لـــ DSA لأفضلية الطالب؛ وقد يوقع أو لا يوقع DSA أي نتائج فيما بعد.

وفي حالة أداء DSA لتسلسل، يكون لدي DSA عدد من الخيارات على أساس شكل النتائج المعادة إلى الطالب، أي:

أ) أعد استجابة مركبة (موقعة أو غير موقعة) إلى الطالب؛

ب) أعد مجموعة من استجابتين جزئيتين غير منسقتين أو أكثر (موقعتين أو غير موقعتين) إلى الطالب؛ وفي هذه المجموعة يمكن توقيع صفر أو أعضاء أكثر ويمكن عدم توقيع صفر أو واحد. وفي حالة نتيجة جزئية غير موقعة محينة، قد يكون هذا العضو في الحقيقة نسق استجابة جزئية واحدة أو أكثر غير موقعة كانت قد استقبلتها من DSAs آخرين، وساهم فيها DSA هذا، أو كلاهما.

وعندما يؤدى DSA وصل لمداخل متعلقة، قد يوقع DSA الذي يؤدى الوصل النتيجة.

القسم 6 – إدارة معرفة

23 نظرة شاملة على إدارة معرفة

لتشغيل دليل موزع على نطاق واسع مع درجة مقبولة من الاتساق والأداء، تطلب إجراءات لانشاء المعرفة التي يحتفظ بها DSA وصيانتها وتمديدها. وتستخدم الآليات التالية معا لإدارة معرفة DSA.

أ) *إسنادات تراتبية وتشغيلية تراتبية غير محددة* – تعرف هذه الإجراءات والبروتوكولات هذه في القسمين 24 و25. وتستخدم لإنشاء وصيانة مراجع تابعة ومراجع تابعة غير محددة ومراجع أعلى مباشرة، وكذلك معلومات سابقة سياق لسياقات تسمية. وتنشأ الإسنادات التشغيلية بين DSAs رئيسيين يحتفظون بسياقات تسمية تتعلق تراتبياً كل منها بالآخر باعتباره تابع مباشر لأعلى مباشر. وقد تحدث الإجراءات كتأثير جانبي لتعديل RDN، أو إضافة أو سحب مدخل، يكون أعلاه المباشر لا يحتفظ به في نفس DSA الذي يحتفظ بالمدخل.

ب) *إسنادات تشغيلية لتظليل* – تعرف الإجراءات والبروتوكولات هذه في ISO/IEC 9594-9 | ITU-T X.525. وتستخدم لانشاء وصيانة مراجع معرفة بطريقتين. الأولى، كتأثير جانبي لانشاء (أو انتهاء) اتفاقات تظليل ونقاط نفاذ تضاف (أو تسحب) من **consumerKnowledge** واختيارياً من نعوت تشغيلية **secondaryShadow**. وقد تستخدم هذه المعلومات بواسطة الإجراءات والبروتوكولات التي تمت مناقشتها أعلاه لتحيين مرجع تابع في DSA رئيسي أعلى والمرجع الأعلى المباشر في DSA رئيسي تابع. والثانية، ينشر DISP مراجع المعرفة التي يحتفظ بها DSAs الرئيسيين لـــ DSAs المستهلكين لظل.

ج) *إسناد مرجعي* – إن توزيع إسناد مرجعي هو خاصية لـــ DSP. واستخدامه هو لإنشاء وصيانة الإسنادات المرجعية الموجزة في 2.23.

ملاحظة – إن آليات تدميث وصيانة المرجع الأعلى و**myAccessPoint** هي خارج مدى مواصفة الدليل هذا.

1.23 صيانة مراجع معرفة

يصف هذا القسم الفرعي كيف يستخدم DOP لصيانة النعوت التشغيلية لـــ DSA التي تعبر عن معرفة. ويرد في الملحق E مثال بسيط للعلاقة بين نعوت معرفة والبروتوكولات المستخدمة لصيانتها.

1.1.23 صيانة معرفة مستهلك بواسطة مورد وDSAs رئيسيين

يجرى التعبير عن مرجع مستهلك من خلال قيمة نعت **consumerKnowledge**، يحتفظ به DSA لمورد ظل ومتصاحب مع سابقة سياق لسياق تسمية؛ ومرجع مورد، من خلال قيمة نعت **supplierKnowledge**، يحتفظ به DSA لمستهلك ظل ومتصاحب أيضاً مع سابقة سياق لسياق تسمية. ويحتفظ بكل من النعتين في DSEs من نمط **cp**. وتخلق قيمة كل واحد من النعتين عند إنشاء Shadow Operational Binding وتحين تعديل Shadow Operational Binding.

يمكن أن يحصل DSA لمورد على معلومات لبناء قيم نعت **secondaryShadows** إذا كان مكون **secondaryShadows** الاختياري لـــ **ShadowingAgreementInfo** مع مستهلك هو **TRUE**. وفي هذه الحالة، عندما يكتشف DSA لمستهلك أن مجموعة من DSAs تحتفظ بنسخ من منطقة مستنسخة قابلة للاستخدام العام (مستهلكين أو بدورهم، مستهلكين لمستهلكين وما إلى ذلك، حتى أقل مستوى من تظليل ثانوي قد ينفذ) قد تغيرت (بواسطة إضافة أو تعديل أو اكتشاف نقاط نفاذ) تنقل المعلومات الجديدة هذه (مجموعة من **SupplierAndConsumers**) بواسطة عملية **modifyOperationalBinding**، كما ورد في ISO/IEC 9594-9 | ITU-T X.525.

يقوم DSA لمورد بصيانة نعته **secondaryShadows** المتصاحب مع سابقة سياق كما يلي:

أ) يمكن استخدام مجموعة من **SupplierAndConsumers** مستقبلة من مستهلك بواسطة عملية **modifyOperationalBinding** لإنشاء أو استبدال قيم النعت. ويمثل مكون مورد **SupplierAndConsumers** نقطة النفاذ لـــ DSA لمستهلك (أو لمستهلكيها وما إلى ذلك يعتمد على أقل مستوى من تظليل ثانوي)؛ مكون المستهلكين ومجموعة مستهلك لمستهلكين (أو مستهلكيهم وما إلى ذلك يعتمد على أقل مستوى من تظليل ثانوي).

ب) إن كل مستهلك يوفر لمورده عملية **modifyOperationalBinding** تحتوى على مجموعة من **SupplierAndConsumers**، تشمل القيم التالية: قيم نعته **secondaryShadows** والقيم المبنية جديداً. وتبني هذه القيمة باستخدام نقطة نفاذها، **myAccessPoint**، (مثل مكون المورد)، وقيم نقاط نفاذ المستهلك، المحتوية في نعت **consumerKnowledge**، الذي يمثل مستهلكين يحتفظون بظلال قابلة للاستخدام العام (مثل مكون مستهلكين).

يسمح الاستخدام المتكرر لهذا الإجراء لـــ DSA رئيسي لسياق تسمية بمعرفة جميع DSAs لمستهلك ظل ثانوي محتفظين بمناطق مستنسخة قابلة للاستخدام العام مشتقة من سياق تسمية. وتتاح هذه المعلومات لصيانة مراجع تابعة وتابعة غير محددة وأعلى مباشرة.

2.1.23 صيانة معرفة تابعة وعليا مباشرة في DSAs رئيسيين

يجرى التعبير عن مرجع تابع من خلال قيمة نعت **specificKnowledge** محتفظ به في DSE من نمط **subr** بواسطة DSA محتفظ بسياق تسمية أعلى مباشر لذلك المرجعي؛ ومرجع أعلى مباشر، من خلال قيمة نعت **specificKnowledge** محتفظ به في DSE من نمط **immSupr** بواسطة DSA محتفظ بسياق تسمية أعلى مباشر لذلك المرجعي. وتنشأ قيمة كل من هذه النعوت في DSAs الأعلى والرئيسيين المباشرين عند إنشاء HOB، وتحين عند تعديل HOB.

يوفر DSA رئيسي تابع لـــ DSA رئيسي أعلى معلومات لبناء مرجع تابع عبر مكون **accessPoints** لمعلمة **SubordinateToSuperior** التي تنقل إلى أعلى في DOP. ويحدد المعلومات المتضمنة في **accessPoints** بواسطة قيم نعوت يحتفظ بها DSA تابع كما يلي:

أ) تستخدم قيمة نعت **myAccessPoint** (المحتفظ بها في جذر DSE) لتشكل عنصراً في **accessPoints** مع **category** لها نفس قيمة **master**.

ب) تستخدم قيم **consumerKnowledge** و**secondaryShadows** (المحتفظ بهما في سابقة سياق تابع لـــ DSE) لتشكل عنصراً إضافياً في **accessPoints** مع **category** لها نفس قيمة **shadow**.

يوفر DSA رئيسي أعلى لـــ DSA رئيسي تابع معلومات لبناء مرجعه الأعلى المباشر عبر مكون **contextPrefixInfo** لمعلمة **SuperiorToSubordinate** التي تنقل إلى تابع في DOP. وهذا المكون هو قيمة نمط **SEQUENCE OF Vertex**، المحتوي على تتابع عناصر تتطابق مع مسير من جذر DIT إلى سابقة سياق تابع. وبالنسبة لأحد هذه العناصر، متطابق مع سابقة سياق لسياق تسمية لأعلى مباشر، يكون المكون الاختياري **accessPoints** محيناً. ويحتفظ DSA التابع بهذه المعلومات باعتبارها نعت **specificKnowledge** في DSE، من نمط **immSupr**، المتطابق مع هذا العنصر لـــ **contextPrefixInfo**. وتحدد المعلومات المتضمنة في **accessPoints** بواسطة DSA أعلى قيم النعوت التي يحتفظ بها DSA أعلى كما يلي:

أ) تستخدم قيمة نعت **myAccessPoint** (المحتفظ بها في جذر DSE) لتشكل عنصراً في **accessPoints** مع **category** لها نفس قيمة **master**.

ب) تستخدم قيم **consumerKnowledge** و**secondaryShadows** (المحتفظ بهما في سابقة سياق أعلى لـــ DSE) لتشكل عنصراً اضافياً في **accessPoints** مع **category** لها نفس قيمة **shadow**.

ملاحظة – إن نقاط النفاذ المتطابقة فقط مع DSAs لمستهلك المستقبلين لمناطق مستنسخة قابلة للاستخدام العام ينبغي أن تختار من قبل DSAs أعلى وتابع من نعوتها **consumerKnowledge** لادراجها في **accessPoints**. وتضمن إجراءات بناء **secondaryShadows** أن نقاط النفاذ هذه تعرف DSAs الظل الذين يحتفظون بمناطق مستنسخة قابلة للاستخدام العام.

3.1.23 صيانة معرفة تابعة وأعلى مباشرة في DSAs لمستهلكين

إن DSA لمستهلك ظل متعاقد مع مورده لاستقبال معرفة أعلى مباشرة وتابعة متصاحبة مع وحدة استنساخ، هو في الواقع، يتعاقد للحصول على مراجع أعلى مباشرة وتابعة يقوم بصيانتها DSA لمورد ظل عبر DISP.

ملاحظة – بالنسبة لبعض مواصفات وحدات استنساخ، قد يكون من الضروري لـــ DSA لمستهلك أن يتعاقد لاستقبال **extendedKnowledge** لكي يزوده مورده بمعرفة تابعة.

2.23 طلب إسناد مرجعي

لتحسين أداء نظام الدليل، يمكن التوسع في المجموعة المحلية لإسنادات مرجعية باستخدام عمليات دليل عادية. وإذا دعم DSA DSP، قد يطلب DSA آخر (الذي يدعم أيضاً DSP) أن يعيد مراجع المعرفة هذه التي تحتوي على معلومات حول مكان موقع سياقات تسمية تتعلق باسم شيء مستهدف لعملية دليل عادية.

إذا ضبط مكون **returnCrossRefs** لـــ **ChainingArguments** على **TRUE**، قد يكون مكون **crossReferences** لـــ **ChainingResults** محيناً، ويتألف من تتابع بنود إسناد مرجعي.

إذا لم يتمكن DSA من تسلسل طلب بـــ DSA التالي، يعاد مرجع إلى DSA الأصلي. وإذا كان مكون **returnCrossRefs** لـــ **ChainingArguments** هو **TRUE**، قد يحتوي المرجع على سابقة سياق إضافية لسياق تسمية يفضله المرجع. ويكون مكون **contextPrefix** غائباً إذا كان المرجع قائماً على أساس مرجع تابع غير محدد. ويكون الإسناد المرجعي، الذي يعيده مرجع، قائماً على أساس معرفة يحتفظ بها DSA الذي ولد المرجع.

وفي كلا الحالتين (سلسلة نتيجة ومرجع) قد تختار سلطة إدارية، من خلال DSA التابع لها، تجاهل طلب إعادة إسنادات مرجعية.

3.23 عدم اتساق معرفة

يتعين على الدليل أن يدعم آليات التحقق من التناسق لضمان درجة معينة من التناسق.

ملاحظة – في بعض الظروف تكون مراجع معرفة دقيقة (ليست غير صالحة بالمعني الوارد أدناه) ولكن غير صالحة ليستخدمها DSA بسبب أن DMD لـــ DSA مرجعي لا يرغب على الاطلاق في أن يتصل به DSA مرجعي (مثل، DSA قد حصل بطريقة ما على إسناد مرجعي بـــ DSA مرجعي) أو لا يرغب الاتصال به بشأن دور معين (مثل، DSA رئيسي لسياق تسمية).

1.3.23 اكتشاف عدم اتساق معرفة

يتفاوت نوع عدم الاتساق واكتشافه للأنماط المختلفة من مراجع معرفة.

أ) *مراجع إسناد وتابعة* – يكون هذا النمط من المراجع غير صالح إذا لم يحتفظ DSA مرجعي بسياق تسمية أو منطقة مستنسخة مشتقة من سياق تسمية مع سابقة سياق متضمنة في المرجع. ويكتشف عدم الاتساق هذا خلال عملية Name Resolution بواسطة تفتيش مكونات **operationProgress** و**referenceType** لـــ **ChainingArguments**.

ب) *مراجع تابعة غير محددة* – يكون هذا النمط من المراجع غير صالح إذا لم يحتفظ DSA مرجعي بسياق تسمية محلي مع سابقة سياق متضمنة في المرجع ناقصاً آخر RDN. ويطبق التحقق من الاتساق كما ورد أعلاه.

ج) *مراجع أعلى* – إن مرجع أعلى غير صالح هو مرجع لا يشكل جزءاً من مسير مرجع بالجذر. وتتم صيانة مراجع أعلى بواسطة وسائل خارجية وهي خارج مدى مواصفة الدليل هذا.

ملاحظة – ليس من الممكن دائماً اكتشاف مرجع أعلى غير صالح.

د) *مراجع أعلى مباشرة* – يكون هذا النمط من المراجع غير صالح إذا لم يحتفظ DSA مرجعي بسياق تسمية أو منطقة مستنسخة مشتقة من سياق تسمية مع سابقة سياق متضمنة في المرجع. وفضلاً عن ذلك، يكون استخدام هذا النمط من المراجع صالحاً فقط عندما يكون لمكون **operationProgress** لـــ **ChainingArguments** قيمة **notStarted** أو **proceeding**. ويكتشف عدم الاتساق هذا خلال عملية Name Resolution بواسطة تفتيش مكونات **operationProgress** و**referenceType** لـــ **ChainingArguments**.

هـ) *مراجع مورد* – يكون هذا النمط من المراجع، الذي يعرف مورد منطقة مستنسخة واختيارياً الرئيسي لسياق تسمية من المنطقة المستنسخة المشتقة، غير صالح إذا لم يسمح DSA مرجعي لمورد الظل لـــ DSA باستخدام المرجع (عندما يكون لمكون **referenceType** لـــ **ChainingArguments** قيمة **supplier**) أو إذا لم يكن DSA مرجعي هو الرئيسي لسياق تسمية (عندما يكون لمكون **referenceType** قيمة **master**). ويكتشف عدم الاتساق هذا خلال Name Resolution ومراحل تقييم عملية لمعالجة عملية بواسطة تفتيش مكون **referenceType** لـــ **ChainingArguments**.

2.3.23 الإبلاغ عن عدم اتساق معرفة

إذا استخدمت سلسلة في أداء طلب لدليل، يكتشف كل عدم اتساق معرفة بواسطة DSA يحتفظ بمرجع معرفة غير صالح، من خلال استقبال **serviceError** مع مشكلة **invalidReference**.

إذا أعاد DSA مرجعاً يكون قائماً على أساس مرجع معرفة غير صالح، يعيد الطالب **serviceError** مع مشكلة **invalidReference** إذا استخدم المرجع. وكيفية انتشار حالة خطأ إلى DSA يخزن مرجع غير صالح ليست في مدى مواصفة الدليل هذا.

3.3.23 معاملة مراجع معرفة غير متسقة

بعد اكتشاف DSA لمرجع غير صالح ينبغي أن يحاول إعادة انشاء اتساق معرفة. فمثلاً، يمكن أن يفعل هذا بواسطة اكتشاف إسناد مرجعي غير صالح أو استبداله بواحد صحيح يمكن الحصول عليه باستخدام آليات **returnCrossRefs**.

إن الطريقة التي يقوم DSA بمناولة مراجع غير صالحة هي مسألة محلية، وخارج مدى مواصفة الدليل هذا.

4.23 مراجع معرفة وسياقات

تكون الأسماء في مراجع معرفة هي أسماء مميزة أولية وقد تشمل قيم مميزة بديلة ومعلومات سياق محتفظ بها في **valuesWithContext** لأي نعت يساهم في أي RDN، كما يرد في 3.9 من التوصية ITU-T X.501 | ISO/IEC 9594-2.

واعتماداً على كيفية الحصول على مرجع معرفة (وخاصة إذا كان DSA للطبعة قبل الثالثة يحتفظ بمرجع أو له جزء من سلسلة يجرى من خلالها الحصول على المرجع)، من الممكن ألا يشمل مرجع المعرفة ذلك جميع الأسماء المميزة البديلة الممكنة. وقد يؤدى هذا إلى اسم لا يتم التعرف عليه باعتباره نفس اسم حامل مرجع المعرفة، مؤدياً إلى خطوات إضافية في استبانة اسم أو

في بعض الحالات، إلى نتائج غير متسقة أو فشل استبانة اسم. إن الاستخدام العام لأسماء مميزة أولية، عندما تعرف، يستمثل قدرة الدليل على تناول متغيرات سياق في أسماء.

24 إسنادات تشغيلية تراتبية

يستخدم إسناد تشغيلي تراتبي ليمثل العلاقة بين DSA اثنين يحتفظان بسياقي تسمية، واحد تابع مباشر للآخر. وفي حالة HOB، يحتفظ DSA الأعلى بمرجع تابع لسياق تسمية يحتفظ به DSA تابع؛ ويحتفظ DSA تابع بمرجع أعلى مباشر لسياق تسمية يحتفظ به DSA أعلى. ويضمن الإسناد التشغيلي أن معلومات معرفة ملائمة يجرى تبادلها وصيانتها بين DSA اثنين بحيث إن كلاهما قادر على السلوك خلال عملية Name Resolution وOperation Evaluation كما وردتا في القسمين 18 و19.

1.24 خاصيات نمط إسناد تشغيلي

1.1.24 التناظر والأدوار

إن نمط إسناد تشغيلي تراتبي هو نمط لا تناظري لإسناد تشغيلي. والدوران في الإسناد لهذا النوع هما:

أ) دور DSA الرئيسي لسياق تسمية أعلى، *superior DSA* (متصاحب مع دور مجرد "A")؛

ب) دور DSA الرئيسي لسياق تسمية تابع، *subordinate DSA* (متصاحب مع دور مجرد "B")

2.1.24 اتفاق

إن اتفاق معلومات متبادل خلال إنشاء إسناد تشغيلي تراتبي هو قيمة **HierarchicalAgreement**. ويحتوى هذا على اسم مميز نسبي لسابقة سياق جديد (مكون **rdn**) واسم مميز لمدخل أعلى مباشر لسياق تسمية جديد (مكون **immediateSuperior**). ويوفر هذه المعلومات DSA الذي بدأ HOB.

```
HierarchicalAgreement ::= SEQUENCE {
        rdn                 [0]  RelativeDistinguishedName,
        immediateSuperior   [1]  DistinguishedName }
```

يكون **rdn** هو RDN الأولى ويكون **immediateSuperior** هو اسم مميز أولي. وتتضمن معلومات سياق وجميع قيم مميزة بديلة في مكون **valuesWithContext** لـــ **AttributeTypeAndDistinguishedValue** لأي RDN كما ورد في ITU-T X.501 | ISO/IEC 9594-2.

3.1.24 المبادر

1.3.1.24 الإنشاء

يمكن تمهيد إنشاء إسناد تشغيلي تراتبي بواسطة أي دور. ويمكن أن يتسبب تدميث بواسطة DSA أعلى في عملية Add Entry مع DSA تابع محدد في تمديد **targetSystem**، أو بواسطة تدخل إداري. ويتسبب التدميث بواسطة DSA تابع (الذي يوصل مدخل موجود محلياً أو شجرة فرعية لـــ DIT عالمية) بواسطة تدخل إداري.

2.3.1.24 التعديل

يمكن تمهيد تعديل إسناد تشغيلي تراتبي بواسطة أي دور. وقد يصدر DSA أعلى التعديل نتيجة لتعديل معلومات سابقة سياق أعلى. ويمكن أن يكون هذا نتيجة لأي عمليات تعديل، أو بواسطة تدخل إداري.

ويمكن لأي من DSA تعديل اتفاق كنتيجة لتعديل RDN لمدخل سابقة سياق لسياق تسمية تابع. ويمهد DSA أعلى هذا التعديل بسبب أن اسماً مميزاً نسبياً يجرى تعديله هو أعلى من DIT، أو بسبب تدخل إداري. ويمهد DSA تابع تعديل بسبب ModifyDN لسابقة سياق أو بسبب تدخل إداري.

ويمكن لأي من DSA تعديل HOB إذا تغيرت معلومات نقطة نفاذ سياق تسميتها.

3.3.1.24 الانتهاء

يمكن تمهيد انتهاء إسناد تشغيلي تراتبي بواسطة أي دور. ويمكن أن يتسبب تدميث بواسطة DSA أعلى من قبل تدخل إداري. والتدميث بواسطة DSA تابع يمكن أن يتسبب فيه إما عملية Remove Entry التي تسحب مدخل سابقة سياق لسياق تسمية تابع أو بواسطة تدخل إداري.

4.1.24 معلمات إنشاء

تختلف معلمات الإنشاء لدوري HOB، DSA أعلى وDSA تابع. وتكون معلمة إنشاء لدور DSA أعلى هي قيمة **SuperiorToSubordinate** ومعلمة دور تابع هي قيمة **SubordinateToSuperior**.

1.4.1.24 معلمة إنشاء DSA أعلى

توفر معلمة الإنشاء التي يصدرها DSA أعلى، قيمة **SuperiorToSubordinate**، لـــ DSA التابع معلومات تتعلق بذروات DIT أعلى من سابقة سياق لسياق تسمية جديد (تشمل مرجع أعلى مباشر) ونعوت اختيارية لمستعمل أعلى وتشغيلية لمدخل سابقة سياق تابع ونسخ لنعوت مستعمل وتشغيلية من المدخل الأعلى المباشر لسابقة السياق الجديد.

```
SuperiorToSubordinate ::= SEQUENCE {
        contextPrefixInfo       [0]  DITcontext,
        entryInfo               [1]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        immediateSuperiorInfo   [2]  SET SIZE (1..MAX) OF Attribute OPTIONAL }
```

يكون **rdn** في **Vertex** أو في **SubentryInfo** هو RDN الأولى، وتتضمن معلومات سياق وجميع القيم المميزة الأخرى في مكونات **AttributeTypeAndDistinguishedValue** لـــ RDN، كما ورد في 3.9 من التوصية ITU-T X.501 | ISO/IEC 9594-2.

1.1.4.1.24 معلومات سابقة سياق

يكون مكون **contextPrefixInfo** لـــ **SuperiorToSubordinate** هو قيمة نمط **DITcontext**، باعتباره تتابع لقيم **Vertex**.

```
DITcontext ::= SEQUENCE OF Vertex

Vertex ::= SEQUENCE {
        rdn             [0]  RelativeDistinguishedName,
        admPointInfo    [1]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        subentries      [2]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL,
        accessPoints    [3]  MasterAndShadowAccessPoints OPTIONAL }
```

يكون مكون **contextPrefixInfo** هو تتابع RDNs الذي يشكل اسماً مميزاً لأعلى مباشر لسابقة سياق جديد، ويتصاحب اختيارياً مع كل RDN (الذي يعطيه مكون **rdn**) معلومات اضافية.

يشير مكون **admPointInfo** لـــ **Vertex** إلى أن ذروة DIT هي نقطة إدارية ويوفر، على الأقل، نعته التشغيلي **administrativeRole**.

يوفر مكون **subentries** لـــ **Vertex** معلومات مدخل فرعي متصاحبة مع نقطة إدارية، وهو مجموعة من قيمة واحدة أو أكثر من **SubentryInfo**. وتتألف كل قيمة **SubentryInfo** من RDN لمدخل فرعي (مكون **rdn**) ونعوت المدخل الفرعي (مكون **info**).

```
SubentryInfo ::= SEQUENCE {
        rdn     [0]  RelativeDistinguishedName,
        info    [1]  SET OF Attribute }
```

يشير مكون **accessPoints** الاختياري لـــ **Vertex** إلى أن الذروة تتطابق مع سابقة سياق لسياق تسمية أعلى مباشر. ويستخدم الأعلى هذا المكون لتوفير التابع الذي تتطلبه المعلومات لمرجعها الأعلى المباشر.

ملاحظة – إن نقطة النفاذ الرئيسية في **accessPoints** هي نفسها كالتي مرت في معلمة **accessPoints** لعمليات Established and Modify Operational Binding.

2.1.4.1.24 معلومات مدخل

إن مكون **entryInfo** الاختياري لـــ **SuperiorToSubordinate** هو مجموعة من النعوت تنشئ محتوى مدخل سابقة سياق جديد.

3.1.4.1.24 معلومات مدخل أعلى مباشر

إن مكون **immediateSuperiorInfo** الاختياري لـــ **SuperiorToSubordinate** هو نسخة لمجموعة نعوت، ولا سيما **objectClass** و**entryACI**، من مدخل أعلى مباشر لسابقة سياق جديد.

ملاحظة – يمكن استخدام هذا المكون من قبل تابع لاستمثال تقييم طلب List الذي يولد **ListResult** فارغة لقاعدة شيء هي أعلى مباشر لسابقة سياق تابع [انظر الملاحظة في 2.2.1.3.19، البند 2].

2.4.1.24 معلمة إنشاء DSA تابع

توفر معلمة الإنشاء التي يصدرها DSA تابع، قيمة **SubordinateToSuperior**، معلومات لـــ DSA أعلى فيما يتعلق بسياق تسمية تابع.

```
SubordinateToSuperior ::=  SEQUENCE {
        accessPoints    [0]  MasterAndShadowAccessPoints OPTIONAL,
        alias           [1]  BOOLEAN DEFAULT FALSE,
        entryInfo       [2]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        subentries      [3]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL }
```

يستخدم مكون **accessPoints** لـــ **SubordinateToSuperior** من قبل التابع ليوفر للأعلى المعلومات المطلوبة لمرجعه التابع.

الملاحظة 1 – إن نقطة النفاذ الرئيسية في **accessPoints** هي نفسها كالتي مرت في معلمة **accessPoints** لعمليات Established and Modify Operational Binding.

يستخدم مكون **alias** لـــ **SubordinateToSuperior** ليشير إلى الأعلى أن سياق تسمية تابع يتألف من مدخل مستعار وحيد.

يتألف مكون **entryInfo** لـــ **SubordinateToSuperior** من نسخة من مجموعة نعوت، ولا سيما **objectClass** و**entryACI**، ولكن أيضاً، إذا انطبق، نعت تشغيلي **administrativeRole**، من مدخل سابقة سياق جديد.

الملاحظة 2 – يمكن استخدام النعتين الأولىين من قبل الأعلى لاستمثال تقييم طلب List أو طلب Search لمستوى واحد تكون قاعدة شيء هي المدخل الأعلى المباشر لسابقة سياق تابع، بينما يستخدم النعت الأخير لتجنب التقدم غير المطلوب لعملية بحث في أو خارج منطقة إدارية محددة لخدمة.

يستخدم مكون **subentries** لـــ **SubordinateToSuperior** من قبل التابع ليمرر مداخل فرعية تحتوي على ACI وصفية إلى الأعلى.

5.1.24 معلمات تعديل

لتعديل HOB، تكون معلمة التعديل لدور الأعلى، **SuperiorToSubordinateModification**، هي **SuperiorToSubordinate**، مع قيد أن مكون **entryInfo** قد لا يكون محيناً؛ وأن دور التابع هو **SubordinateToSuperior**.

```
SuperiorToSubordinateModification ::= SuperiorToSubordinate (
        WITH COMPONENTS { ..., entryInfo ABSENT})
```

وهذه المعلمات متماثلة (مع القيد الوارد أعلاه) لمعلمات الإنشاء المتطابقة وتستخدم لتشير إلى تغيير استحدث للمعلومات الموفرة في معلمات الإنشاء التالية لإنشاء HOB.

إذا مر أي مكون لـــ **SuperiorToSubordinate** (أو **SuperiorToSubordinateModification** تالي) أو **SubordinateToSuperior** بتغيير (مثل، مكون **contextPrefixInfo** لـــ **SuperiorToSubordinate**)، يتوفر المكون المتطابق لمعلمة التعديل (مثل، مكون **contextPrefixInfo** لـــ **SuperiorToSubordinateModification**) بكامله في Modify Operational Binding.

6.1.24 معلمات الانتهاء

لا يوفر أي من الأدوار معلمة عند انتهاء HOB.

7.1.24 تعريف نمط

يعرف إسناد تشغيلي تراتبي معرف شيء معين عند تعريف شيء لمعلومات **hierarchicalOperationalBinding** **OPERATIONAL-BINDING** في 2.24.

2.24 تعريف صنف شيء لمعلومات إسناد تشغيلي

يعرف القسم الفرعي هذا نمط إسناد تشغيلي تراتبي باستخدام نموذج صنف شيء لمعلومات **OPERATIONAL-BINDING** المعرف في ITU-T X.501 | ISO/IEC 9594-2.

```
hierarchicalOperationalBinding OPERATIONAL-BINDING ::= {
        AGREEMENT       HierarchicalAgreement
        APPLICATION CONTEXTS {
                        {directorySystemAC} }
        ASYMMETRIC
            ROLE-A {            -- superior DSA
                ESTABLISHMENT-INITIATOR         TRUE
                ESTABLISHMENT-PARAMETER         SuperiorToSubordinate
                MODIFICATION-INITIATOR          TRUE
                MODIFICATION-PARAMETER          SuperiorToSubordinateModification
                TERMINATION-INITIATOR           TRUE }
            ROLE-B {            -- subordinate DSA
                ESTABLISHMENT-INITIATOR         TRUE
                ESTABLISHMENT-PARAMETER         SubordinateToSuperior
                MODIFICATION-INITIATOR          TRUE
                MODIFICATION-PARAMETER          SubordinateToSuperior
                TERMINATION-INITIATOR           TRUE }
        ID                  id-op-binding-hierarchical }
```

3.24 إجراءات DSA لإدارة إسناد تشغيلي تراتبي

في الإجراءات التالية، يخزن DSE جديد أو علامة (أي، دلالة حالة متصاحبة مع بند معلومات) أنشأها DSA في تخزين مستقر. وعند القيام بذلك، من الممكن لـــ DSA اثنين اتباع الإجراءات أدناه للحفاظ على فهم متسق لمعلمات HOB في وجود اتصالات وفشل نظام طرفي.

في كل من إجراء **establishment** و**modification** الوارد أدناه، يمكن أن يوفر DSA القائم بدور مستجيب (أي، لا يمهد لـــ establishment وmodification) لـــ DSA القائم بدور التمهيد مع معلومات (مثل، نعوت تشغيلية) لا تقبل لسبب أو لآخر. ويمكن لـــ DSA الممهد أن ينهي إسناداً تشغيلياً في هذه الحالات.

1.3.24 إجراء الإنشاء

1.1.3.24 الإنشاء الذي يمهد له DSA أعلى

إذا قام DSA بتقييم عملية Add Entry مع DSA مختلف محدد في تمديد **targetSystem**، ينشئ إسناد تشغيلي تراتبي طبقا للإجراء التالي. وإذا رغب DSA، لأسباب إدارية، أن ينشئ HOB مع DSA تابع، يدعم بروتوكول DOP HOB، تم يتبع الإجراء التالي:

1) ينشئ DSA الأعلى DSE جديد من نمط **subr**، مع اسم مدخل جديد، ويعلم DSE الجديد هذا على أنه *being added*. ويولد DSA الأعلى **bindingID** فريد ويخزنه مع DSE الجديد.

2) يرسل DSA الأعلى عملية Establish Operational Binding إلى DSA تحتوي على المعلمات التالية:

أ) **bindingType** مضبوط على **hierarchicalOperationalBindingID**؛

ب) معلمة إنشاء **SuperiorToSubordinate** مع مكونات **contextPrefixInfo** و**entryInfo** محينة؛ وتكون جميع المعلمات الأخرى اختيارية؛

ج) **HierarchicalAgreement** مع مكون **immediateSuperior** مضبوط على اسم مميز لأعلى مباشر للمدخل الجديد ومكون **rdn** مضبوط على RDN المدخل الجديد؛

د) معلمات **bindingID** و**myAccessPoint** و**valid**، حسب الاقتضاء.

3) إذا قبل DSA التابع العملية، يخلق DSEs المطلوبة من أنماط **glue** و**subentry** و**admPoint** و**rhob** و**immSupr** حسب الاقتضاء، لتمثل **contextPrefixInfo**؛ ويمثل DSE من نمط **cp** و**entry** أو **alias** شيء سابقة سياق جديد أو مدخل مستعار؛ وحسب الاقتضاء، فإن DSE من نمط **rhob** و**entry** يمثل **immediateSuperiorInfo**. ويخزن **bindingID** مع DSE لمدخل سابقة سياق جديد ويعيد معلمة **SubordinateToSuperior** إلى DSA الأعلى.

وإذا رفض DSA تابع العملية، يعيد Operational Binding Error مع قيمة مشكلة ملائمة مضبوطة.

وإذا وجد سياق تسمية فعلا وكانت قيم **bindingID** للسياق الموجود والجديد هي نفسها، يكون DSA التابع قد خلق فعلاً سياق تسمية مطلوب، وفي هذه الحالة يعيد DSA التابع نتيجة إلى الأعلى. وإذا لم تكن النتائج متساوية، يرسل Operational Binding Error مع مشكلة **invalidAgreement**؛ ويعني هذا أن DSA الأعلى له عدم اتساق معرفة دائم يتطلب تصحيحاً من إداري.

4) إذا استقبل DSA الأعلى خطأ، يلغي DSE المعلم من نمط **subr** ويعيد خطأ لعملية Add Entry.

إذا استقبل DSA الأعلى نتيجة، يقوم بسحب العلامة من DSE الذي يمثل **subr** ويعيد نتيجة عملية Add Entry.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA الأعلى الخطوات ابتداءً من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل إنشاء معلق لإسناد تشغيلي تراتبي يكون هو المبادر. وإذا كان الإنشاء هو نتيجة لعملية Add Entry، يجهض الطالب العملية (مثلاً، بواسطة تحرير أو إجهاض تصاحب تطبيق) قبل انتهاء الإنشاء، ويتجاهل DSA الأعلى هذا الحدث ويستكمل الإنشاء (الذي يكون أو لا يكون ناجحاً). وفي هذه الحالة، لا يتم اخطار المستعمل بناتج عملية Add Entry.

الملاحظة 1 – إن تعليم التابع يساعد على الاستعادة ومراقبة التلازم. ولا يتمكن مستعمل آخر من إضافة مدخل يكون معلم فعلاً، ويكرر DSA إنشاء إسناد تشغيلي لجميع التوابع المعلمة بعد فشل.

الملاحظة 2 – ومع الإجراء أعلاه، يكون للمعرفة عدم اتساق عابر. وهي مسألة محلية كيف يعامل DSA الأعلى العمليات غير المتعلقة التي تقرأ مرجعاً تابعاً بينما يكون معلماً.

2.1.3.24 الإنشاء الذي يمهد له DSA تابع

قد يمهد DSA تابع لإسناد تشغيلي تراتبي. وينتج هذا من رغبة إداري في توصيل شجرة فرعية لمداخل محتفظ بها في DSA بنقطة معينة في DIT عالمية. وفي هذه الحالة، ينشئ DSA التابع HOB طبقاً للإجراء التالي:

1) يكون لدى DSA التابع إما DSE من نمط **cp** كجزء من سياق تسمية موجود أو يخلق واحداً جديداً. ويعلم DSE على أنه *being added*، ويولد **bindingID** فريد ويخزنه مع سابقة سياق DSE.

2) يرسل DSA التابع عملية Establish Operational Binding إلى DSA الأعلى تحتوي على المعلمات التالية:

أ) **bindingType** مضبوط على **hierarchicalOperationalBindingID**؛

ب) معلمة إنشاء **SubordinateToSuperior**، حسب الاقتضاء؛

ج) **HierarchicalAgreement** مع مكون **immediateSuperior** مضبوط على اسم مميز لأعلى مباشر للمدخل الجديد ومكون **rdn** مضبوط على RDN المدخل الجديد؛

د) معلمات **bindingID** و**myAccessPoint** و**valid**، حسب الاقتضاء

وإذا رفض DSA التابع العملية، يعيد Operational Binding Error مع قيمة مشكلة ملائمة مضبوطة.

3) يتحقق DSA الأعلى من أنه الرئيسي للأعلى المباشر لمدخل سابقة سياق جديد أو يعيد **operationalBindingError** مع مشكلة **roleAssignment**.

4) يتحقق DSA الأعلى من أن RDN المطلوب لسابقة السياق الجديد لا يوجد قيد الاستعمال. وإذا لم توجد مواءمة RDN باستخدام معلومات محتفظ بها محلياً، ولكن DSE الأعلى المباشر هو من نمط **nssr**، يتبع الإجراء 5.1.19. وإذا اكتشف عدم مواءمة RDN باستخدام هذا الإجراء، يخلق DSA الأعلى DSE من نمط **subr** ويخزن **bindingID** معه، ويعيد نتيجة.

إذا وجد مرجع تابع مع RDN هذا، تجرى مقارنة قيمتي **bindingID**. وإذا كانتا متساويتان، تعاد نتيجة. ولا تحتوي معلمة **SuperiorToSubordinate** التي أعادها DSA الأعلى على مكون **entry**. وإذا كانت قيمتا **bindingID** غير متساويتين، يرسل **operationalBindingError** مع مشكلة **invalidAgreement**؛ ويعني هذا أن DSA الأعلى له عدم اتساق معرفة دائم يتطلب تصحيحاً من إداري.

إذا وجدت مواءمة RDN بواسطة استكشاف NSSR، يرسل **operationalBindingError** مع مشكلة **invalidAgreement**؛ ويعني هذا أن DSA الأعلى له عدم اتساق معرفة دائم يتطلب تصحيحاً من إداري.

5) إذا استقبل DSA التابع خطأ، يلغي سابقة قياس جديد لـــ DSE وعلامته. وهي مسألة محلية لتحديد مصير معلومات المدخل التي اشتقت منه سابقة قياس DSE.

إذا استقبل DSA التابع نتيجة، يضيف DSEs الضرورية من أنماط **glue** و**subentry** و**admPoint** و**rhob** و**immSupr** حسب الاقتضاء، لتمثل **contextPrefixInfo**؛ ويمثل DSE من نمط **rhob** و**entry immediateSuperiorInfo**. وتسحب علامة سابقة قياس DSE.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA الأعلى الخطوات ابتداءً من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل إنشاء معلق لإسناد تشغيلي تراتبي يكون هو المبادر.

2.3.24 إجراء تعديل

تعرف الإجراءات التالية لتعديل HOB مهّد له الإجراء المفصل في 1.3.24.

1.2.3.24 إجراء تعديل يمهد له الأعلى

يمكن تنفيذ هذا الإجراء كنتيجة لعمليات تعديل، كما ورد في 1.19، أو نتيجة لتدخل إداري (مثل، نقل تغييرات لمعلمات **myAccessPoint** أو **agreement** أو **valid** لـــ HOB). وأيضاً، إذا كتشف DSA الأعلى تغييرات في مكونات **contextPrefixInfo** أو **immediateSuperiorInfo** لقيمة **SuperiorToSubordinate** التي وردها إلى DSA التابع.، وينشر المعلومات الجديدة على DSA الأعلى المستخدم للإجراء التالي:

1) علّم DSE من نمط **subr** على أنه *being added*، وإذا كان هذا التعديل نتيجة تعديل على RDN لمدخل سابقة سياق تابع، يضاف DSE من نمط **subr** ويعلّم على أنه *being added*.

2) ينتج DSA الأعلى قيمة **bindingID** جديدة من القيمة الموجودة بواسطة زيادة مكون **version**. وباستخدام **bindingID** الجديد هذا، يرسل عملية Modify Operational Binding إلى DSA التابع مع معلمة تعديل **SuperiorToSubordinateModification**.

3) يتحقق DSA التابع من مكون **identifier** لـــ **bindingID**. وإذا لم يكن له اتفاق مع الأعلى، أو إذا كان مكون **version** أقل من صيغة HOB، يعيد **operationalBindingError** مع مشكلة **invalidAgreement**.

4) يمكن أن يقبل DSA التابع تعديلاً على HOB، ويعدل أو يعيد بناء DSEs الممثلين لمعلومات سابقة سياق، ويحين مكون **version** لـــ **bindingID** ويعيد نتيجة. وبديلاً عن ذلك، قد يعيد خطأ ثم ينهي الاتفاق.

5) إذا استقبل DSA الأعلى نتيجة، يكتمل التعديل. وإذا كان هذا التعديل نتيجة تعديل على RDN مدخل سابقة سياق تابع، وكان DSE الذي له نمط **subr** ومعلّم على أنه *being added*، تسحب علامته، ويحذف DSE القديم المعلّم على أنه *being modified*، وإذا لم يكن، تسحب علامة *being modified*.

إذا استقبل DSA الأعلى خطأ، يكون التعديل قد فشل. وتسحب علامة *being modified*، وإذا كان هذا التعديل نتيجة تعديل على RDN لمدخل سابقة سياق تابع، يسحب DSE الجديد من نمط **subr** والمعلّم على أنه *being added*. وإذا لم يكن، فإن التدابير التي تتخذ هي خارج مدى مواصفة الدليل هذا.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA الأعلى الخطوات ابتداءً من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل تعديل معلق لإسناد تشغيلي تراتبي يكون هو المبادر. وإذا كان هذا التعديل نتيجة

تعديل ModifyDN لتعديل RDN لمدخل سابقة سياق تابع، ويجهض الطالب العملية (مثلاً، بواسطة تحرير أو إجهاض تصاحب تطبيق) قبل انتهاء التعديل، يتجاهل DSA الأعلى هذا الحدث ويستكمل التعديل (الذي يكون أو لا يكون ناجحاً). وفي هذه الحالة، لا يتم إخطار المستعمل بناتج عملية ModifyDN.

2.2.3.24 إجراء تعديل يمهد له التابع

يمكن تنفيذ هذا الإجراء كنتيجة لتدخل إداري (مثل، نقل تغييرات لمعلمات **myAccessPoint** أو **agreement** أو **valid** لـــ HOB). وأيضاً، إذا إكتشف DSA تابع تغييرات في قيمة **SubordinateToSuperior** التي وردها إلى DSA الأعلى، ينشر المعلومات الجديدة على DSA الأعلى المستخدم للإجراء التالي:

1) علّم DSE من نمط **cp** على أنه *being modified*.

2) ينتج DSA الأعلى قيمة **bindingID** جديدة من القيمة الموجودة بواسطة زيادة مكون **version**. وباستخدام **bindingID** الجديد هذا، يرسل عملية Modify Operational Binding إلى DSA الأعلى مع معلمة تعديل **SubordinateToSuperior**.

3) يتحقق DSA الأعلى من مكون **identifier** لـــ **bindingID**. وإذا لم يكن له اتفاق مع التابع، أو إذا كان مكون **version** أقل من صيغة HOB، يعيد **operationalBindingError** مع مشكلة **invalidAgreement**.

4) يمكن أن يقبل DSA الأعلى تعديلاً على HOB، ويعدل DSEs الممثل لمرجع التابع ويعيد نتيجة. وبديلاً عن ذلك، قد يعيد خطأ ثم ينهي الاتفاق.

وبالإضافة إلى ذلك، إذا كان يتعين على DSE الأعلى لـــ DSE (من نمط **subr**) إعادة تسميته، يكون من نمط **nssr**، ويتبع DSA الإجراء المعرف في 5.1.19 (Modify Operations وNSSRs) لضمان أن الاسم الجديد للمدخل هو غير مبهم، قبل الاستجابة لطلب تعديل HOB.

5) إذا استقبل DSA التابع نتيجة، يكتمل التعديل ويسحب العلامة. وإذا استقبل خطأ، فإن التدابير التي تتخذ هي خارج مدى مواصفة الدليل هذا.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA التابع الخطوات ابتداء من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل تعديل معلق لإسناد تشغيلي تراتبي يكون هو المبادر.

3.3.24 إجراء انتهاء

تعرف الإجراءات التالية لإنهاء HOB الذي مهّد له الإجراء المفصل في 1.3.24.

1.3.3.24 انتهاء يمهد له DSA أعلى

يمهد DSA الأعلى انتهاء إسناد تشغيلي تراتبي كنتيجة فقط لتدخل إداري. ويتبع الإجراء التالي:

1) يعلم DSA الأعلى DSE الممثل لمرجع تابع *being deleted*، بحيث إن المرجع التابع لم يعد مستخدماً خلال Name Resolution.

2) يرسل DSA الأعلى عملية Terminate Operational Binding من أجل إسناد تشغيلي تراتبي لـــ DSA التابع. ويحذف الأعلى مكون **version** لـــ **bindingID**.

3) عندما يستقبل DSA التابع Terminate Operational Binding، يحذف أي معلومات عن إسناد تشغيلي تراتبي ويرسل نتيجة، ما لم يكن مكون **identifier** لـــ **bindingID** غير معروف، وفي هذه الحالة يعاد

operationalBindingError مع مشكلة **invalidID**. وهي مسألة محلية لتحديد مصير أي معلومات مدخل متصاحبة مع سياق تسمية تابع.

4) إذا استقبل DSA الأعلى نتيجة أو **operationalBindingError** مع مشكلة **invalidID**، يحذف DSE المعلم *being deleted*، الممثل لمرجع تابع متصاحب مع إسناد تشغيلي تراتبي ويحذف أي معلومات عن إسناد تشغيلي.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA الأعلى الخطوات ابتداء من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل انتهاء معلق لإسناد تشغيلي تراتبي يكون هو المبادر.

2.3.3.24 انتهاء يمهد له DSA تابع

إن الانتهاء الذي يمهّد له DSA التابع يمكن أن تتسبب فيه عملية Remove Entry التي تسحب المدخل الأخير في سياق تسمية تابع أو مدخل سابقة سياق أو كنتيجة لتدخل إداري. ويتبع الإجراء التالي:

1) يعلم DSA التابع DSE لسابقة سياق لسياق تسمية *being deleted*.

2) يرسل DSA التابع عملية Terminate Operational Binding من أجل إسناد تشغيلي تراتبي لـــ DSA الأعلى. ويحذف التابع مكون **version** لـــ **bindingID**.

3) عندما يستقبل DSA التابع Terminate Operational Binding، يحذف DSE الذي يمثل مرجع تابع متصاحب مع إسناد تشغيلي تراتبي، ويلغي أي معلومات عن إسناد تشغيلي ويرسل نتيجة، ما لم يكن مكون **identifier** لـــ **bindingID** غير معروف، وفي هذه الحالة يعاد **operationalBindingError** مع مشكلة **invalidID**.

4) إذا استقبل DSA التابع نتيجة أو **operationalBindingError** مع مشكلة **invalidID**، يحذف أي معلومات عن إسناد تشغيلي.

ملاحظة – إن مصير معلومات مدخل لسياق تسمية هو مسألة محلية لـــ DSA تابع. ونظراً لأن تسمية (أي، تحريك) سياق تسمية لا تسمح به عملية Modify DN، يمكن لإداري، مثلاً، إنهاء HOB واختيار سابقة سياق لسياق تسمية إعادة وصله بآخر كجزء من DIT (أي، إنشاء HOB جديد).

إذا حدث أي فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA التابع الخطوات ابتداء من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل انتهاء معلق لإسناد تشغيلي تراتبي يكون هو المبادر.

4.24 إجراءات عمليات

إن العمليات التي يمكن أن تنفذ بطريقة تعاونية لإسناد تشغيلي تراتبي هي التي تعرف في سياق تطبيق **directorySystemAC**.

إن الإجراءات التي تتضمن DSA في إسناد تشغيلي تراتبي التي تتبع هي المعرفة في القسم 16 حتى 22.

5.24 استخدام سياقات تطبيق

لإنشاء أو تعديل أو إنهاء إسناد تشغيلي تراتبي باستخدام البروتوكول وإجراءات معيار الدليل هذا، يستخدم DSA سياق تطبيق **operationalBindingManagementAC**.

25 إسناد تشغيلي تراتبي غير محدد

يستخدم إسناد تشغيلي تراتبي غير محدد ليمثل العلاقة بين DSA اثنين يحتفظان بسياقي تسمية، واحد تابع مباشر للآخر. وفي حالة NHOB، يحتفظ DSA الأعلى بمرجع تابع غير محدد لسياق تسمية يحتفظ به DSA التابع؛ ويحتفظ DSA التابع بمرجع

أعلى مباشر لسياق تسمية يحتفظ به DSA الأعلى. ويضمن الإسناد التشغيلي أن معلومات معرفة ملائمة يجرى تبادلها وصيانتها بين DSA اثنين بحيث أن كلاهما قادر على السلوك خلال عملية استبانة اسم وتقييم عملية كما ورتا في القسمين 18 و19.

1.25 خاصيات نمط إسناد تشغيلي

1.1.25 التناظر والأدوار

إن نمط إسناد تشغيلي تراتبي هو نمط لا تناظري لإسناد تشغيلي. والدوران في الإسناد لهذا النوع هما:

أ) دور DSA الرئيسي لسياق تسمية أعلى، *superior DSA* (متصاحب مع دور مجرد "A")؛

ب) دور DSA الرئيسي لسياق تسمية تابع، *subordinate DSA* (متصاحب مع دور مجرد "B").

2.1.25 اتفاق

إن اتفاق معلومات متبادل خلال إنشاء إسناد تشغيلي تراتبي غير محدد هو قيمة **NonSpecificHierarchicalAgreement** يحتوى فقط على اسم مميز لمدخل أعلى مباشر لسابقة تسمية جديدة (مكون **immediateSuperior**). ويوفر هذه المعلومات DSA الذي مهد NHOB.

```
NonSpecificHierarchicalAgreement ::= SEQUENCE {
        immediateSuperior     [1]   DistinguishedName }
```

ملاحظة – إن كيفية تحديد DSA التابع بأن الاسم لسياق تسمية جديد هو غير مبهم هي خارج مدى هذه التوصية/المعيار الدولي هذا. ويكون الاسم غير مبهم إذا عين صحيحاً من قبل سلطة تسمية ذات علاقة وإذا لم يحتفظ DSA بنفس اسم مدخل رئيسي.

3.1.25 المبادر

1.3.1.25 الإنشاء

يمكن تمهيد إنشاء إسناد تشغيلي تراتبي غير محدد بواسطة أي دور لـــ DSA تابع. ويتسبب تدميث بواسطة DSA تابع (الذي يوصل مدخلاً واحداً أو أكثر موجود محلياً أو شجرة فرعية بـــ DIT عالمية) بواسطة تدخل إداري.

2.3.1.25 التعديل

يمكن تمهيد تعديل إسناد تشغيلي تراتبي غيرمحدد بواسطة أي دور. وقد يصدر DSA الأعلى التعديل كنتيجة لتعديل معلومات سابقة سياق أعلى. ويمكن أن يكون هذا نتيجة لأي عمليات تعديل، أو بواسطة تدخل إداري.

ويمكن لأي من DSA تعديل NHOB إذا تغيرت معلومات نقطة نفاذ سياق تسميتها (أو أحد سياقات تسمية تابع مباشر في حالة دور التابع).

3.3.1.25 الانتهاء

يمكن تمهيد انتهاء إسناد تشغيلي تراتبي بواسطة أي دور. ويمكن أن يتسبب تدميث بواسطة DSA أعلى من قبل تدخل إداري. والتدميث بواسطة DSA تابع يمكن أن تتسبب فيه إما عملية Remove Entry التي تسحب مدخل سابقة سياق نهائي يحتفظ بتابع مباشر لتابع مكون **immediateSuperior** للاتفاق أو بواسطة تدخل إداري.

4.1.25 معلمات إنشاء

تكون معلمات الإنشاء التي يصدرها DSA الأعلى، قيمة **NHOBSuperiorToSubordinate**،مساوية لمعلمة إنشاء HOB المتطابقة، باستثناء أن يكون مكون **entryInfo** غائباً.

```
NHOBSuperiorToSubordinate ::= SuperiorToSubordinate (
        WITH COMPONENTS { ..., entryInfo ABSENT})
```

تكون معلمات الإنشاء التي يصدرها DSA التابع، قيمة **NHOBSubordinateToSuperior**،مساوية لمعلمة إنشاء HOB المتطابقة، باستثناء أن تكون مكونات **alias** و**entryInfo** غائبة.

```
NHOBSubordinateToSuperior ::= SEQUENCE {
    accessPoints      [0]  MasterAndShadowAccessPoints OPTIONAL,
    subentries        [3]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL }
```

5.1.25 معلمات تعديل

هذه المعلمات متماثلة لمعلمات الإنشاء المتطابقة وتستخدم لتشير إلى تغييرات تحدث للمعلومات الموفرة في معلمات الإنشاء التالية لإنشاء NHOB.

إذا مر أي مكون لـــ **NHOBSuperiorToSubordinate** أو **NHOBSubordinateToSuperior** بتغيير (مثل، مكون **contextPrefixInfo** لـــ **NHOBSuperiorToSubordinate**)، يتوفر المكون المتطابق لمعلمة التعديل (مثل، مكون **contextPrefixInfo** لـــ **NHOBSuperiorToSubordinate**) بكامله في Modify Operational Binding.

6.1.25 معلمات الانتهاء

لا يوفر أي من الأدوار معلمة عند انتهاءNHOB .

7.1.25 تعريف نمط

يعرف إسناد تشغيلي تراتبي غير محدد معرف بواسطة معرف شيء معين عند تعريف شيء لمعلومات **nonSpecificHierarchicalOperationalBinding OPERATIONAL-BINDING** في 2.25.

2.25 تعريف صنف شيء لمعلومات إسناد تشغيلي

يعرف القسم الفرعي هذا نمط إسناد تشغيلي تراتبي غير محدد باستخدام نموذج صنف شيء لمعلومات **OPERATIONAL-BINDING** المعرف في ISO/IEC 9594-2 | ITU-T X.501.

```
nonSpecificHierarchicalOperationalBinding OPERATIONAL-BINDING ::= {
        AGREEMENT       NonSpecificHierarchicalAgreement
        APPLICATION CONTEXTS {
                        { directorySystemAC } }
        ASYMMETRIC
            ROLE-A {            -- superior DSA
                ESTABLISHMENT-PARAMETER     NHOBSuperiorToSubordinate
                MODIFICATION-INITIATOR      TRUE
                MODIFICATION-PARAMETER      NHOBSuperiorToSubordinate
                TERMINATION-INITIATOR       TRUE }
            ROLE-B {            -- subordinate DSA
                ESTABLISHMENT-INITIATOR     TRUE
                ESTABLISHMENT-PARAMETER     NHOBSubordinateToSuperior
                MODIFICATION-INITIATOR      TRUE
                MODIFICATION-PARAMETER      NHOBSubordinateToSuperior
                TERMINATION-INITIATOR       TRUE }
        ID                  id-op-binding-non-specific-hierarchical }
```

3.25 إجراءات DSA لإدارة إسناد تشغيلي تراتبي غير محدد

في الإجراءات التالية، كما في الإجراءات الواردة في 3.24، يخزن DSE جديد أو علامة أنشأها DSA بطريقة تخزين مستقرة.

في كل من إجراء establishment وmodification الوارد أدناه، يمكن أن يوفر DSA القائم بدور مستجيب (أي، لا يمهد لـــ establishment وmodification) لـــ DSA القائم بدور التمهيد مع معلومات (مثل، نعوت تشغيلية) لا تقبل لسبب أو لآخر. ويمكن لـــ DSA الممهد أن ينهي إسناداً تشغيلياً في هذه الحالات.

1.3.25 إجراء الإنشاء

يمكن أن يمهد DSA التابع فقط اسناداً تشغيلياً تراتبياً. وينتج هذا من رغبة إداري في توصيل شجرة فرعية واحدة أو أكثر لمداخل محتفظ بها في DSA بنقطة معينة في DIT عالمية. وفي هذه الحالة، ينشئ DSA التابع NHOB طبقاً للإجراء التالي:

1) يكون لـــ DSA تابع إما DSE من نمط **cp**، كجزء من سياق تسمية موجود أو يخلق واحداً جديداً، ويعلم DSE على أنه *being added* ويولد **bindingID** فريد ويخزنه مع سابقة سياق DSE.

2) يرسل DSA التابع عملية Establish Operational Binding إلى DSA الأعلى تحتوي على المعلمات التالية:

   أ) **bindingType** مضبوط على **nonSpecificHierarchicalOperationalBindingID**؛

   ب) معلمة إنشاء **NHOBSubordinateToSuperior**، حسب الاقتضاء؛

   ج) **NonSpecificHierarchicalAgreement** مع مكون **immediateSuperior** مضبوط على اسم مميز لأعلى مباشر للمدخل الجديد ؛

   د) معلمات **bindingID** و**myAccessPoint** و**valid**، حسب الاقتضاء.

3) يتحقق DSA الأعلى من أنه الرئيسي للأعلى المباشر لمدخل سابقة سياق جديد أو يعيد **operationalBindingError** مع مشكلة **roleAssignment**.

4) يضيف DSA الأعلى DSE من نمط **nssr** (ومعلومات نعت **nonSpecificKnowledge**) إلى DSE الأعلى المباشر للمدخل الجديد، يخزن **bindingID** معه، ويعيد نتيجة.

5) إذا استقبل DSA التابع خطأ، يلغي سابقة قياس جديد لـــ DSE وعلامته. وهي مسألة محلية لتحديد مصير معلومات المدخل التي اشتقت منه سابقة قياس DSE.

   إذا استقبل DSA التابع نتيجة، يضيف DSEs الضرورية من أنماط **glue** و**subentry** و**admPoint** و**rhob** و**immSupr** حسب الاقتضاء، لتمثل **contextPrefixInfo**؛ ويمثل DSE من نمط **rhob** و**entry** **immediateSuperiorInfo**. وتسحب علامة سابقة قياس DSE.

   إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA التابع الخطوات ابتداء من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل إنشاء معلق لإسناد تشغيلي تراتبي يكون هو المبادر.

2.3.25 إجراء تعديل

إذا كشف DSA الأعلى أي تغييرات في معلومات **NHOBSuperiorToSubordinate** قد وردها إلى DSA تابع في إسناد تشغيلي تراتبي غير محدد، ينشر المعلومات المتغيرة على DSA التابع. وإذا كان NHOB قد نشأ باستخدام الإجراء 1.3.25،

فيعدل طبقاً للإجراءات المعرفة لتعديل إسناد تشغيلي تراتبي في 1.2.3.24 (مع **NHOBSuperiorToSubordinate** المبدل بـ **SuperiorToSubordinateModification**).

وبالمثل، إذا كشف DSA التابع أي تغييرات في معلومات **NHOBSubordinateToSuperior** قد وردها إلى DSA الأعلى، ينشر التغييرات على DSA الأعلى. وإذا كان NHOB قد نشأ باستخدام الإجراء 1.3.25، فيعدل طبقاً للإجراءات المعرفة لتعديل إسناد تشغيلي تراتبي في 2.2.3.24 (مع **NHOBSubordinateToSuperior** المبدل بـ **SubordinateToSuperior**).

3.3.25 إجراء انتهاء

تعرف الإجراءات التالية لإنهاء NHOB الذي نشأ باستخدام الإجراء 1.3.25.

1.3.3.25 انتهاء يمهد له DSA أعلى

يمهد DSA الأعلى انتهاء إسناداً تشغيلياً تراتبياً كنتيجة فقط لتدخل إداري. ويتبع الإجراء التالي:

1) يعلم DSA الأعلى القيمة المتطابقة مع DSA التابع في نعت **nonSpecificKnowledge** المحتفظ به في DSE لمدخل أعلى مباشر على أنه *being deleted*.

2) يرسل DSA الأعلى عملية Terminate Operational Binding من أجل NHOB لـ DSA التابع. ويحذف الأعلى مكون **version** لـ **bindingID**.

3) عندما يستقبل DSA التابع Terminate Operational Binding، يحذف أي معلومات عن NHOB ويرسل نتيجة، ما لم يكن مكون **identifier** لـ **bindingID** غير معروف، وفي هذه الحالة يعاد **operationalBindingError** مع مشكلة **invalidID**. وهي مسألة محلية لتحديد مصير أي معلومات مدخل متصاحبة مع سياق تسمية تابع.

4) إذا استقبل DSA الأعلى نتيجة أو **operationalBindingError** مع مشكلة **invalidID**، يحذف قيمة نعت **nonSpecificKnowledge** المعلم *being deleted*، الممثل لمعلومات نقطة نفاذ متصاحبة مع NHOB ويحذف أي معلومات عن إسناد تشغيلي. وإذا كان هذا هو القيمة الأخيرة لنعت **nonSpecificKnowledge**، يسحب نعت **nonSpecificKnowledge** وDSE من نمط **nssr** من DSE.

إذا حدث فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA الأعلى الخطوات ابتداء من الخطوة 2 حتى تستقبل نتيجة أو خطأ لكل انتهاء معلق لـ NHOB يكون هو المبادر.

2.3.3.25 انتهاء يمهد له DSA تابع

إن الانتهاء الذي يمهد له DSA التابع يمكن أن تتسبب فيه عملية Remove Entry التي تسحب المدخل الأخير في سياق تسمية تابع أو مدخل سابقة سياق أو السياق التابع الأخير المحتفظ به DSA التابع أو كنتيجة لتدخل إداري. ويتبع الإجراء التالي:

1) يعلم DSA التابع DSE لسابقة سياق لسياق تسمية *being deleted*.

2) يرسل DSA التابع عملية Terminate Operational Binding من أجل إسناد تشغيلي تراتبي لـ DSA الأعلى. ويحذف التابع مكون **version** لـ **bindingID**.

3) عندما يستقبل DSA الأعلى Terminate Operational Binding، يلغي قيمة نعت **nonSpecificKnowledge** الذي يمثل معلومات نقطة نفاذ متصاحبة مع NHOB، ويلغي أي معلومات حول إسناد تشغيلي، ويسحب نعت **nonSpecificKnowledge** وDSE لنمط **nssr** من DSE الأعلى المباشر إلى سياق تسمية تابع (إذا كانت

القيمة الملغاة هي القيمة الأخيرة لنعت **nonSpecificKnowledge**) ويرسل نتيجة، ما لم يكون مكون **identifier** لـــ **bindingID** غير معروف، وفي هذه الحالة يعاد **operationalBindingError** مع مشكلة **invalidID**.

4) إذا استقبل DSA التابع نتيجة أو **operationalBindingError** مع مشكلة **invalidID**، يلغي أي معلومات حول الإسناد التشغيلي. وهي مسألة محلية لتقرير مصير أي معلومات مدخل متصاحبة مع سياق تسمية التابع.

إذا حدث أي فشل (مثل، اتصالات أو نظام طرفي)، يكرر DSA التابع الخطوات ابتداءً من الخطوة 2 حتى يستقبل نتيجة أو خطأ لكل إنهاء معلق لـــ NHOB يكون فيه هو المبادر.

4.25 إجراءات للعمليات

إن العمليات التي يمكن أن تنفذ بطريقة تعاونية لإسناد تشغيلي تراتبي غير محدد هي التي تعرف في سياق تطبيق **directorySystemAC**.

إن الإجراءات التي يشترك فيها DSA في إسناد تشغيلي تراتبي غير محدد التي تتبع هي المعرفة في القسم 16 حتى 22.

5.25 استخدام سياقات تطبيق

لإنشاء أو تعديل أو انهاء إسناد تشغيلي تراتبي غير محدد باستخدام البروتوكول وإجراءات معيار الدليل هذا، يستخدم DSA سياق تطبيق **operationalBindingManagementAC**.

الملحق A
ASN.1 لعمليات موزعة

(يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

يشمل هذا الملحق جميع تعاريف أنماط وقيم ASN.1 الواردة في مواصفة الدليل هذا في شكل وحدة ASN.1 **DistributedOperations**.

```
DistributedOperations  {joint-iso-itu-t ds(5) module(1) distributedOperations(3) 5}
DEFINITIONS ::=
BEGIN

-- EXPORTS All --
-- تصدر الأنماط والقيم المعرفة في هذه الوحدة لاستخدامها في وحدات ASN.1 الواردة في مواصفات الدليل، وللاستخدام
--في التطبيقات الأخرى التي تستخدم للنفاذ إلى خدمات الدليل. ويمكن أن تستخدمها تطبيقات أخرى لأغراضها الخاصة،
--ولكن لا يقيد هذا التمديدات والتعديلات المطلوبة للحفاظ على خدمة الدليل أو تحسينها.

IMPORTS

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

        basicAccessControl, commonProtocolSpecification, directoryAbstractService, enhancedSecurity,
        informationFramework,selectedAttributeTypes, serviceAdministration, upperBounds
            FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}

        DistinguishedName, Name, RDNSequence
            FROM InformationFramework informationFramework

        MRMapping, SearchRuleId
            FROM ServiceAdministration serviceAdministration

        AuthenticationLevel
            FROM BasicAccessControl basicAccessControl

        OPTIONALLY-PROTECTED{ }
            FROM EnhancedSecurity enhancedSecurity

-- from ITU-T Rec. X.511 | ISO/IEC 9594-3

        abandon, addEntry, CommonResults, compare, directoryBind, list,
        modifyDN, modifyEntry, read, referral, removeEntry, search, SecurityParameters
            FROM DirectoryAbstractService directoryAbstractService

-- from ITU-T Rec. X.519 | ISO/IEC 9594-5

        ERROR, id-errcode-dsaReferral, OPERATION
            FROM CommonProtocolSpecification commonProtocolSpecification
```

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

        DirectoryString{}, PresentationAddress, ProtocolInformation, UniqueIdentifier
            FROM SelectedAttributeTypes selectedAttributeTypes

        ub-domainLocalID, ub-labeledURI
            FROM UpperBounds upperBounds;

-- parameterized type for deriving chained operations --

chained { OPERATION : operation } OPERATION  ::=  {
        ARGUMENT  OPTIONALLY-PROTECTED {
            SET {
                chainedArgument ChainingArguments,
                argument        [0]   operation.&ArgumentType } }
        RESULT    OPTIONALLY-PROTECTED {
            SET {
                chainedResult   ChainingResults,
                result          [0]   operation.&ResultType } }
        ERRORS{ operation.&Errors EXCEPT referral | dsaReferral }
        CODE    operation.&operationCode }

--  bind and unbind operations --

dSABind             OPERATION  ::=  directoryBind

--dSAUnbind          OPERATION  ::=  directoryUnbind

--  chained operations --

chainedRead         OPERATION  ::=  chained { read }

chainedCompare      OPERATION  ::=  chained { compare }

chainedAbandon      OPERATION  ::=  abandon

chainedList         OPERATION  ::=  chained { list }

chainedSearch       OPERATION  ::=  chained { search }

chainedAddEntry     OPERATION  ::=  chained { addEntry }

chainedRemoveEntry  OPERATION  ::=  chained { removeEntry }

chainedModifyEntry  OPERATION  ::=  chained { modifyEntry }

chainedModifyDN     OPERATION  ::=  chained { modifyDN }

-- errors and parameters --

dsaReferral   ERROR  ::=  {
        PARAMETER OPTIONALLY-PROTECTED {
            SET {
                reference       [0]   ContinuationReference,
                contextPrefix   [1]   DistinguishedName OPTIONAL,
                COMPONENTS OF CommonResults } }
        CODE        id-errcode-dsaReferral }

--  common arguments and results  --

ChainingArguments  ::=  SET {
        originator                  [0]   DistinguishedName OPTIONAL,
        targetObject                [1]   DistinguishedName OPTIONAL,
        operationProgress           [2]   OperationProgress
                                    DEFAULT { nameResolutionPhase notStarted },
        traceInformation            [3]   TraceInformation,
        aliasDereferenced           [4]   BOOLEAN DEFAULT FALSE,
        aliasedRDNs                 [5]   INTEGER OPTIONAL,
                            -- only present in first edition systems
        returnCrossRefs             [6]   BOOLEAN DEFAULT FALSE,
```

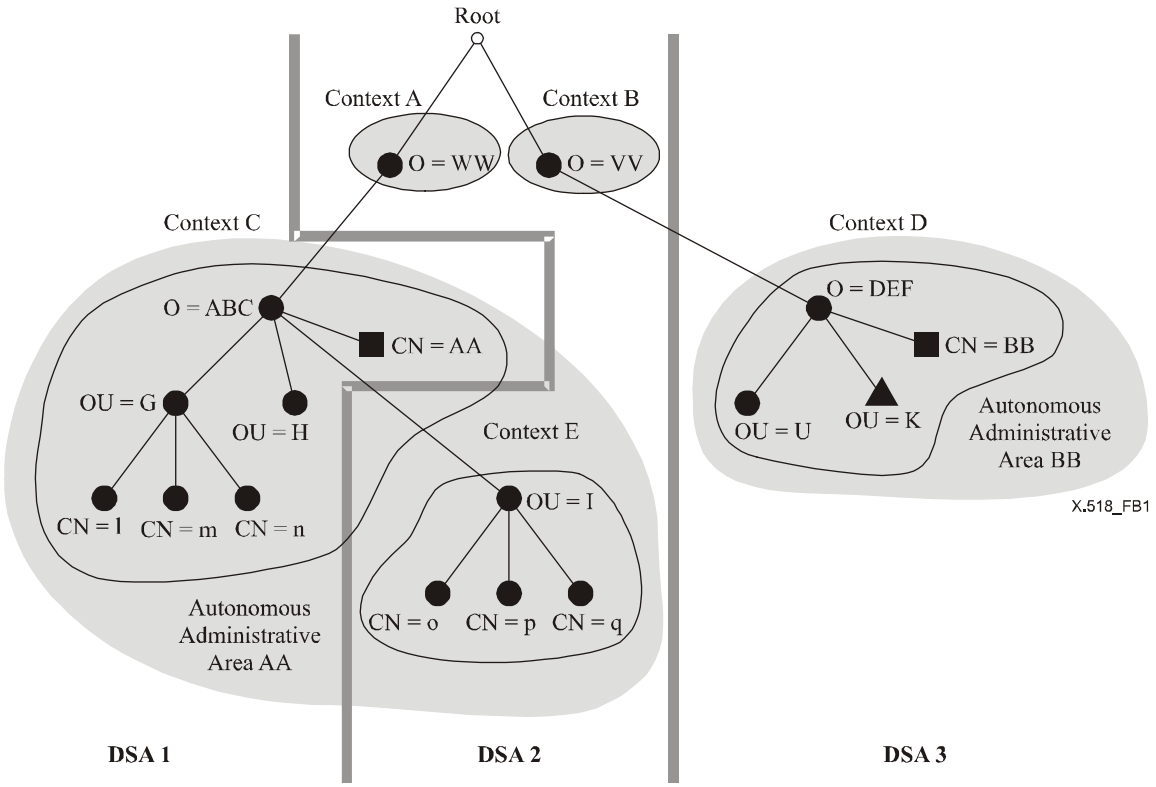
```
        referenceType           [7]  ReferenceType DEFAULT superior,
        info                    [8]  DomainInfo OPTIONAL,
        timeLimit               [9]  Time OPTIONAL,
        securityParameters      [10] SecurityParameters DEFAULT { },
        entryOnly               [11] BOOLEAN DEFAULT FALSE,
        uniqueIdentifier        [12] UniqueIdentifier OPTIONAL,
        authenticationLevel     [13] AuthenticationLevel OPTIONAL,
        exclusions              [14] Exclusions OPTIONAL,
        excludeShadows          [15] BOOLEAN DEFAULT FALSE,
        nameResolveOnMaster     [16] BOOLEAN DEFAULT FALSE,
        operationIdentifier     [17] INTEGER OPTIONAL,
        searchRuleId            [18] SearchRuleId  OPTIONAL,
        chainedRelaxation       [19] MRMapping     OPTIONAL,
        relatedEntry            [20] INTEGER  OPTIONAL,
        dspPaging               [21] BOOLEAN DEFAULT FALSE,
        nonDapPdu               [22] ENUMERATED { ldap (0) } OPTIONAL,
        streamedResults         [23] INTEGER OPTIONAL,
        excludeWriteableCopies  [24] BOOLEAN DEFAULT FALSE }

Time ::= CHOICE {
        utcTime         UTCTime,
        generalizedTime GeneralizedTime }

DomainInfo  ::=  ABSTRACT-SYNTAX.&Type

ChainingResults ::= SET {
        info                [ 0] DomainInfo OPTIONAL,
        crossReferences     [1]  SEQUENCE SIZE (1..MAX) OF CrossReference OPTIONAL,
        securityParameters  [2]  SecurityParameters DEFAULT { },
        alreadySearched     [3]  Exclusions OPTIONAL }

CrossReference ::= SET {
        contextPrefix   [0] DistinguishedName,
        accessPoint     [1] AccessPointInformation }

OperationProgress ::= SET {
        nameResolutionPhase     [0] ENUMERATED {
            notStarted  (1),
            proceeding  (2),
            completed   (3) },
        nextRDNToBeResolved     [1] INTEGER OPTIONAL }

TraceInformation ::= SEQUENCE OF TraceItem

TraceItem ::= SET {
      dsa               [0] Name,
      targetObject      [1] Name  OPTIONAL,
      operationProgress [2] OperationProgress }

ReferenceType ::= ENUMERATED {
        superior                (1),
        subordinate             (2),
        cross                   (3),
        nonSpecificSubordinate  (4),
        supplier                (5),
        master                  (6),
        immediateSuperior       (7),
        self                    (8),
        ditBridge               (9) }

AccessPoint ::= SET {
      ae-title          [0] Name,
      address           [1] PresentationAddress,
      protocolInformation [2] SET SIZE (1..MAX) OF ProtocolInformation OPTIONAL,
      labeledURI        [6] LabeledURI OPTIONAL }

LabeledURI ::= DirectoryString{ub-labeledURI}

MasterOrShadowAccessPoint ::= SET {
      COMPONENTS OF     AccessPoint,
```

```
        category              [3]  ENUMERATED {
                master              (0),
                shadow              (1) } DEFAULT master,
        chainingRequired      [5]  BOOLEAN  DEFAULT FALSE }

MasterAndShadowAccessPoints  ::=  SET SIZE (1..MAX) OF MasterOrShadowAccessPoint

AccessPointInformation  ::=  SET {
        COMPONENTS OF               MasterOrShadowAccessPoint ,
        additionalPoints      [4]   MasterAndShadowAccessPoints OPTIONAL }

DitBridgeKnowledge  ::=  SEQUENCE {
        domainLocalID               DirectoryString{ub-domainLocalID}  OPTIONAL,
        accessPoints          MasterAndShadowAccessPoints }

Exclusions  ::=  SET SIZE (1..MAX) OF RDNSequence

ContinuationReference  ::=  SET {
        targetObject                [0]   Name,
        aliasedRDNs                 [1]   INTEGER OPTIONAL, -- only present in first edition systems
        operationProgress           [2]   OperationProgress,
        rdnsResolved                [3]   INTEGER OPTIONAL,
        referenceType               [4]   ReferenceType,
        accessPoints                [5]   SET OF AccessPointInformation,
        entryOnly                   [6]   BOOLEAN DEFAULT FALSE,
        exclusions                  [7]   Exclusions OPTIONAL,
        returnToDUA                 [8]   BOOLEAN DEFAULT FALSE,
        nameResolveOnMaster         [9]   BOOLEAN DEFAULT FALSE }

END  -- DistributedOperations
```

الملحق B
مثال لاستبانة اسم موزع

(لا يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

إن الشكل B.1 هو مثال لكيفية استخدام استبانة اسم موزع لمعالجة طلبات دليل مختلفة. ويقوم المثال على أساس DIT افتراضية وتشكيل (تشكيلات) DSA المتوافق الوارد في الملحق O (وضع نماذج لمعرفة) من التوصية ITU-T X.501/ISO/IEC 9594-2 وتم استنساخها هنا للتسهيل.



الشكل B.1 – DIT افتراضية متقابلة مع ثلاثة DSA

مع افتراض أسلوب سلسلة للانتشار، تعالج الطلبات الموجه إلى ASN.1 التالية كما يلي:

1) طلب مع اسم مميز {C = WW, O = ABC, OU = G, CN = l}

– تتوائم استبانة اسم بنجاح مع كل RDN في الاسم المستهدف مع DSEs المحتفظ بها ASN.1، حتى يحدد موقع DSE المستهدف.

2) طلب مع اسم مميز {C = WW, O = JPR}

– يتواءم إجراء Name Resolution في DSA 1 مع DSE C= WW، ويكون غير قادر على مزيد من المواءمة. وعند هذه النقطة، يجد DSA 1 مرجعين محتملين لمساعدته على الاستمرار: واحد هو مرجع

immSupr في DSE C= WW، والآخر هو مرجع **supr** في جذر DSE. وفي المثال الافتراضي هذا، يشير كلاهما إلى DSA 2. ولهذا، يسلسل الطلب بـــ DSA 2.

- في DSA 2، يتواءم إجراء **Name Resolution** مع DSE C= WW، ويكون غير قادر على مزيد من المواءمة. وفي هذه الحالة، يكون DSE C= WW هو **cp** و**entry**، وDSA 2 هو DSA الرئيسي لهذا المدخل، ولا يكون هناك **nssr** عند C= WW، ولهذا يكون DSA 2 قادراً على تحديد عدم وجود مثل هذا الاسم في الدليل. ويعاد **nameError** مع مشكلة **noSuchObject**.

3) طلب مع اسم مميز {C = VV, O = DEF, OU = K}

- لا يتمكن إجراء **Name Resolution** في DSA 1 من التوائم مع أي DSE. ويكون المرجع الوحيد المتاح هو مرجع **supr** في جذر DSE، الذي يشير إلى DSA 2. ولهذا يتسلسل الطلب بـــ DSA 2.

- في DSA 2، يتواءم **Name Resolution** مع DSE C= VV، ثم DSE O= DEF ويكون غير قادر على مزيد من المواءمة. ونظراً لأن DSE O= DEF موجود على أنه من نمط **subr**، يستخدم مرجع المعرفة المحدد، الذي يشير إلى DSA 3، ويتسلسل الطلب بـــ DSA 3.

- وفي DSA 3، يوائم **Name Resolution** اسم الشيء المستهدف بالكامل، ويجد أن DSE المحدد موقعه هو من نمط **alias**. ومع افتراض ابدال مستعارين في هذه الحالة، يُبنى اسم جديد باستخدام **aliasedEntryName** المتضمن في DSE التوائم. ويعود DSA 3 إلى دخول إجراء **Name Resolution** ليستمر.

الملحق C
الاستخدام الموزع للاستيقان

(لا يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

1.C موجز

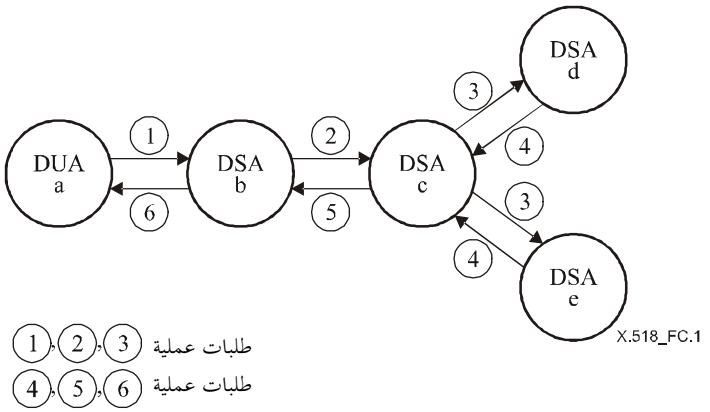
يعرف نموذج الأمن هذا في القسم 17 من ISO/IEC 9594-2 | ITU-T X.501. وما يلي هو موجز للنقاط الأساسية للنموذج:

أ) يدعم الاستيقان القوي، بواسطة توقيع الطلب والنتيجة والأخطاء في DSP.

ب) يدعم تجفير الطلب والنتيجة والأخطاء في DSP.

يصف هذا الملحق كيفية تحقيق هذه في الدليل الموزع. ويستفيد من المصطلحات والترميز المعرف في ISO/IEC 9594-8 | ITU-T X.509.

2.C نموذج حماية موزعة



الشكل 1.C – حماية موزعة

يوضح الشكل 1.C النموذج الذي يستخدم لتحديد إجراءات حماية موزعة. ويعرف النموذج تتابع تدفق المعلومات للحالة العامة لعملية List أو Search. وتعتبر العملية على أنها تنشأ من DUA 'a' مستشهدا بشيء مستهدف يوجد في DSA 'c' عند إجراء عملية، وتتضمن DSAs 'b' و'c' و'd' و'e'.

يتصل DUA 'a' مبدئياً بأي DSA ('b DSA) الذي لا يحتفظ بشيء مستهدف، ولكنه قادر على المرور، عبر سلسلة، إلى DSA ('c DSA) المحتفظ بشيء مستهدف. وإذا كان جميع DSAs تعملون بأسلوب مرجع، يكون النموذج بسيطاً جداً، ويساوي كل DSA/DSA بدالة، على أساس الحماية، التفاعل بين DUA 'a' و'b DSA.

1.2.C نوعية الحماية

توضع نوعية الحماية المستخدمة خلال حياة تصاحب تطبيق خلال عملية Bind للدليل. وتؤكد سياسة نظام مستوى الحماية الذي يلتزم به DUA وDSA. وDIRQOP هو صنف شيء لمعلومات يمكن استخدامه لتحديد نوعية الحماية التي تتصاحب مع كل عملية (طلب أو نتيجة أو أخطاء). ويحمل DUA صنف شيء لمعلومات **DIRQOP** في **DirectoryBindArgument**،

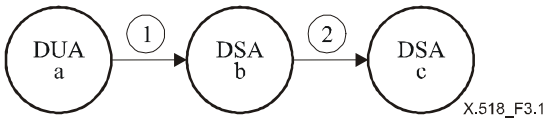
ويقبل DSA هذا المستوى من الحماية في **DirectoryBindResult**. ويمكن استخدام نوعية الحماية لتوفير الأنماط التالية للحماية: موقعة أو مجفرة أو موقعة ومجفرة.

3.C عمليات متسلسلة موقعة

إذا دعمت عمليات متسلسلة موقعة رقمياً، يكون DUA مسؤولاً عن التحقق من أن DSA قد أعاد التوقيعات الرقمية في نتيجة List أو Search. ويتطلب هذا أن يكون DUA قادراً على التحقق من التوقيعات الرقمية من أكثر من DSA واحد إذا استخدمت بيئة موزعة لتوليد نتائج List أو Search. وتطابق نتائج List أو Search هي مسؤولية DUA. ولا ينبغي على DSAs دمج هذه النتائج نيابة عن DUA. وفي بعض الحالات، قد يستقبل DUA معلومات من DSAs مختلفين يدعم كل منهم مستويات مختلفة من الاستيقان والتوقيعات الرقمية. ويصبح قرار DUA ما إذا كان يستخدم أم لا المعلومات المعادة إذا كان التوقيع غير صالح.

1.3.C متغيرات موقعة متسلسلة

إذا وقع متغير DAP من قبل DUA، ينبغي أن يبقي التوقيع طوال حياة الطلب. ويمكن التحقق من هذا التوقيع ويستخدمه DSAs عند أداء Access Control Verification. وإذا قرر DSA أن الطلب يحتاج إلى تسلسله بـــ DSA آخر من أجل المعالجة، ينبغي أن يتضمن الطلب الموقع لـــ DUA مع متغيرات سلسلة ضرورية. وإذا كان DSA ينوى دعم عمليات DSP موقعة (DSA-to-DSA) تستخدم تفويضات DSA لتوقيع DSP **ChainingArguments** وينبغي أن يظل توقيع DUA طوال طلب DAP الأصلي.

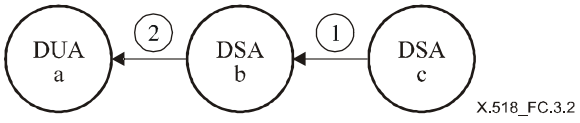


(1) DUA 'a' user signs DAP request

(2) DSA 'b' signs DSP Chaining Argument
(DAP request signed by DUA 'a' user)

2.3.C نتائج موقعة متسلسلة

إذا رغب مستعمل DUA في استقبال نتائج موقعة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **SIGNED**. وينبغي أن يكون لـــ DSA عن بعد القدرة على التشكيل لارسال **ChainingResults** موقعة رقمياً. ويمكن لـــ DSA عن بعد أن يوقع اختيارياً نتيجة DAP و**ChainingResults** DAP، وبالتالي يدعم توقيعات من طرف إلى طرف. ويكون DSA 'b' مسؤولاً عن التحقق من توقيعات DSP لـــ DSA عن بعد، ويكون DUA 'a' مسؤولاً عن التحقق من DAP Result Signature لـــ DSA.



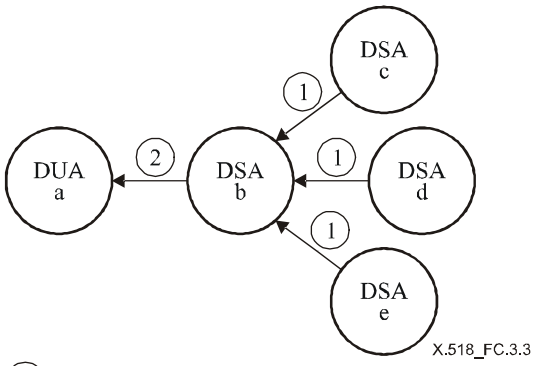
(1) DSA 'c' signs DSP Chaining Result and signs DAP Result

(2) DSA 'b' returns DAP result signed by DSA 'c'

3.3.C دمج نتائج موقعة لـــ List أو Search

يتطلب هذا أن يكون DUA قادراً على التحقق من التوقيعات الرقمية من أكثر من DSA واحداً إذا استخدمت بيئة موزعة لتوليد نتائج List أو Search. وتطابق نتائج عمليات List أو Search هي مسؤولية DUA. ولا ينبغي على DSAs دمج هذه النتائج نيابة عن مستعمل DUA. وفي بعض الحالات، قد يستقبل DUA معلومات من DSAs مختلفين يدعم كل منها

مستويات مختلفة من الاستيقان والتوقيعات الرقمية. ويصبح قرار DUA ما إذا كان يستخدم أم لا المعلومات المعادة إذا كان التوقيع غير صالح.



(1) DSA 'c', 'd', 'e' sign DSP Chaining Result (DAP result signed by DSA 'c', 'd', 'e')

(2) DSA 'b' returns partial DAP result signed by DSA 'c', 'd' and 'e', DSA 'b' does not merge the DAP results

ملاحظة – يمكن أيضاً توقيع بروتوكول DSP (DSA-to-DSA) أوتجفيره وتوقيعه وتجفيره.

4.3.C طلب سلسلة متعددة

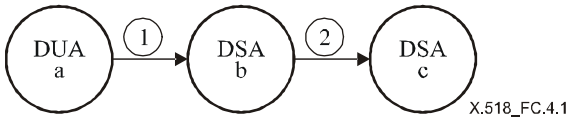
إذا قرر DSA أن طلب DAP يحتاج إلى تسلسله بـــ DSAs متعددين آخرين، يمكن أن يسلسل تعددياً الطلب إما بالتوازي أو تتبعياً. ويوصف أسلوبان للتجزئة: NSSR أو تجزئة طلب. وفي تجزئة NSSR، يرسل DSA طلباً مماثلاً لـــ DSAs معرفين آخرين. وفي طلب تجزئة، يرسل DSA طلباً جزئياً تتابعياً (من الممكن مختلف) إلى كل من DSAs آخرين.

4.C عمليات متسلسلة مجفرة

إذا دعم التجفير، تحتاج حماية مماثلة إلى توفيرها بين كل من مكونات الدليل. ويطلب التقابل، خارج مدى هذه المواصفة، للاتفاق فيما يتعلق بتماثل السياسات.

1.4.C تجفير من نقطة إلى نقطة (DUA->DSA or DSA->DSA) لطلب

إذا أراد مستعمل DUA أن يجفر طلب DAP، يمكن أن يحدث التجفير فقط على أساس من نقطة إلى نقطة. ويجفر DUA طلب DAP لـــ DSA 'b' ؛ ومع ذلك، لا يعرف مستعمل DUA ما إذا كان الطلب أم لا سيسلسل في النهاية بـــ DSA عن بعد لمعالجته. ويجفر DSA 'b' الطلب ويحاول أن ينجز الطلب. وإذا قرر DSA 'b' أن الطلب ينبغي تسلسله بـــ DSA آخر (DSA 'c') لمعالجته، يجفر DSA 'b' العمليات المتسلسلة لـــ DSA 'c'. ويدل على اختيار حماية من نقطة إلى نقطة لطلب DSP واستجاباته (متغيرات ونتائج عملية متسلسلة) بواسطة **dirqop** القائم بين DSA 'b' وDSA 'c' في DSP Bind.



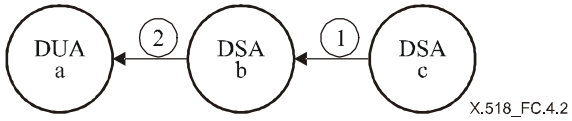
(1) DUA 'a' user encrypts DAP request for DSA 'b'

(2) DSA 'b' encrypts DSP chained operation argument

2.4.C تجفير من نقطة إلى نقطة (DUA->DSA or DSA->DSA) لنتيجة

إذا رغب مستعمل DUA في استقبال نتائج أو أخطاء مجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في Arguments عملية متسلسلة ليعكس **DIRQOP** في **BindArgument** DAP. وينبغي على DSA (DSA 'c') أن يكون له قدرة تشكيل لارسال نتائج عملية متسلسلة مجفرة. وفي هذا السيناريو، يحدد نظام

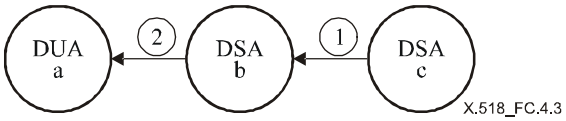
'c' DSA أن باستطاعته انجاز الطلب، ويولد نتيجة DAP ونتائج عملية متسلسلة لـــ DSP. ويمكن تحقيق تجفير من نقطة إلى نقطة بواسطة 'c' DSA المجفر لنتائج عملية متسلسلة DSP لـــ 'b' DSA. ويمكن لـــ 'b' DSA أن يجفر نتائج عملية متسلسلة لـــ DSP ويجفر نتيجة لمستعمل 'a' DUA. ويوفر هذا تجفيراً من نقطة إلى نقطة للنتيجة. ويكون 'a' DUA مسؤولاً عن تجفير نتيجة DAP لـــ ('b' DSA) DSA محلي.



(1) DSA 'c' encrypts DSP chaining operation Result
(2) DSA 'b' encrypts DAP Result for DUA 'a'

3.4.C تجفير من نقطة إلى نقطة لنتيجة DAP وتجفير من نقطة إلى نقطة لنتيجة متسلسلة لـــ DSP

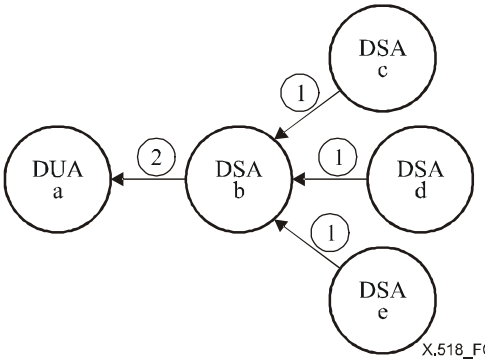
إذا رغب مستعمل 'a' DUA في استقبال نتائج أو أخطاء مجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في Arguments عملية متسلسلة ليعكس **DIRQOP** في DAP Bind. وينبغي على 'c' DSA عن بعد أن يكون له قدرة تشكيل لارسال نتائج عملية متسلسلة مجفرة. وفي هذا السيناريو، يحدد نظام 'c' DSA أن باستطاعته إنجاز الطلب، ويولد تجفير من نقطة إلى نقطة لنتيجة DAP (لمستعمل DUA) وتجفير من نقطة إلى نقطة لنتيجة عملية متسلسلة لـــ DSP. ويمكن تجفير من نقطة إلى نقطة بواسطة 'c' DSA لأنه يعرف من هو مستعمل 'a' DUA المقصود. ويمكن تحقيق نتائج عملية متسلسلة لـــ DSP بواسطة 'c' DSA المجفر لنتائج عملية متسلسلة DSP لـــ 1 DSA. ويمكن لـــ 'b' DSA أن يجفر DSP وترحيل نتيجة DAP المجفرة إلى مستعمل 'a' DUA. ويكون 'a' DUA مسؤولاً عن تجفير نتيجة DAP التي يستقبلها من 'c' DSA عبر 'b' DSA.



(1) DSA 'c' encrypts the DSP chained operation result for DSA 'b'; this includes the DAP result from DSA 'c' that was encrypted for DUA 'a'
(2) DSA 'b' returns the DAP result that was encrypted by DSA 'c' for DUA 'a'

4.4.C دمج نتائج Search/List (دمج مع إعادة تجفير بواسطة DSA 1)

إذا رغب مستعمل 'a' DUA في استقبال نتائج أو أخطاء List أو Search مجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في Arguments عملية متسلسلة ليعكس **DIRQOP** في DAP Bind. وقد يختار DSA محلي ('b' DSA) تسلسل متعدد لطلب list/search لـــ DSAs آخرين عديدين (سواء بالتوازي أو تتابعياً). وينبغي على DSAs ('c' و'd' و'e') عن بعد أن يكون لهم القدرة على تشكيل لارسال نتائج list/search متسلسلة مجفرة. وفي هذا النموذج، ينجز كل من DSAs ('c' و'd' و'e') الطلب ويولد نتائج DAP ونتائج عملية متسلسلة مجفرة لـــ DSP. وتنقل نتائج العملية المتسلسلة التي ولدها DSAs ('c' و'd' و'e') إلى 'b' DSA. ويستقبل 'b' DSA كل من نتائج عملية متسلسلة، ويجفر النتائج وينسق أو يدمج النتائج في نتيجة مشتركة واحدة. ثم يجفر إلى 'b' DSA نتيجة list/search المشتركة الجديدة هذه ويرسلها إلى مستعمل 'a' DUA. ويتحقق تجفير من نقطة إلى نقطة بواسطة DSAs عن بعد لتجفير نتائج عملية متسلسلة لـــ DSP لأجل 'b' DSA وبواسطة تجفير 'b' DSA لنتيجة DAP لمستعمل 'a' DUA. ويكون DUA مسؤولا عن تجفير نتيجة DAP مدمجة واحدة.



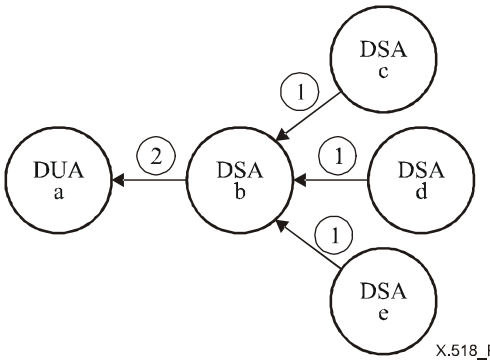
(1) DSA 'c', 'd', 'e' encrypt the DSP chained operation results (including DAP result)

(2) DSA 'b' decrypts the DSP chained operation results from DSA 'c', DSA 'd' and DSA 'e', then merges the DAP results and re-encrypts the DAP result for DUA 'a'

5.4.C دمج غير مسموح لنتائج Search/List

(ليس هناك دمج بواسطة DSA 'b' متوفر لتجفير من طرف إلى طرف لنتيجة DAP list/search)

إذا رغب مستعمل DUA 'a' في استقبال نتائج أو أخطاء list أو search مجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في Arguments عملية متسلسلة ليعكس **DIRQOP** في DAP Bind. وقد يختار DSA محلي تسلسل متعدد لطلب list/search لـــ DSAs آخرين عديدين (سواء بالتوازي أو تتابعياً). وينبغي على DSAs ('c' و'd' و'e') أن يكون لهم القدرة على تشكيل لارسال نتائج list/search متسلسلة مجفرة. وفي هذا السيناريو، ينجز كل DSAs ('c' و'd' و'e') الطلب ويولد نتائج DAP مجفرة (لمستعمل DUA 'a') ونتائج عملية متسلسلة مجفرة لـــ DSP (لـــ DSA 'b'). وتنقل نتائج العملية المتسلسلة التي ولدها DSAs ('c' و'd' و'e') إلى DSA 'b'. ويستقبل DSA 'b' كل نتائج عملية متسلسلة، ويجفر نتائج عملية متسلسة DSP ولا يؤدى أي نمط من النسق أو يدمج نتائج. ويرحل DSA 'b' نتائج list/search (التي جفرها 'c' و'd' و'e') ويرسلها إلى مستعمل DUA 'a' دون تعديل. ويتحقق تجفير من نقطة إلى نقطة بواسطة DSAs عن بعد لتجفير نتائج list/search لـــ DAP لمستعمل DUA 'a' ويتحقق تجفير من نقطة إلى نقطة بواسطة DSAs عن بعد لتجفير نتائج عملية متسلسة DSP لـــ DSA 'b'. ويكون DUA 'a' مسؤولاً عن تجفير نتائج DAP list/search المعادة.



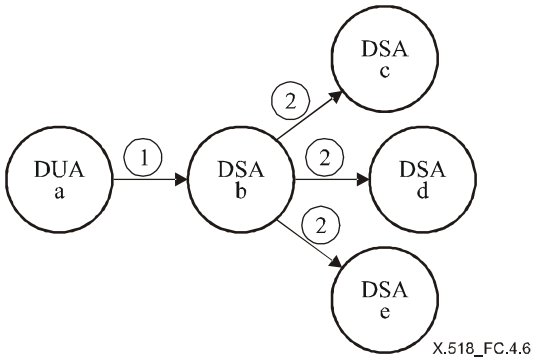
(1) DSA 'c', 'd', 'e' encrypt the DSP chained operation Results for DSA 'b'; this includes those that have been encrypted for the DUA 'a' user

(2) DSA 'b' decrypts the DSP chained operation Results from DSA 'c', DSA 'd', and DSA 'e', then relays the DAP results (which were encrypted by 'c', 'd' and 'e' for DUA 'a') without decrypting or merging them to DUA 'a'

6.4.C سلسلة متعددة لطلب DAP باستخدام مفتاح تجفير (مفتاح شبكة)

إذا رغب مستعمل DUA 'a' في استقبال نتائج أو أخطاء مجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال

SecurityParameters.ProtectionRequest في Arguments عملية متسلسلة ليعكس **DIRQOP** في DAP Bind. وقد يختار DSA محلي ('DSA 'b) تسلسل متعدد لطلب list/search لـــ DSAs آخرين عديدين (سواء بالتوازي أو تتابعياً). ويمكن تشكيل ('DSA 'b) DSA محلي لدعم مفتاح تجفير أو مفتاح شبكة. ومفتاح شبكة هو مفتاح تجفير تناظري يتقاسمه جميع DSAs في السلسلة. وباستخدام مفتاح شبكة، يحتاج 'DSA 'b فقط إلى تجفير طلب متسلسل واحد. ويعرف كل DSAs عن بعد مفتاح الشبكة وهو قادر على تجفير متغير عملية متسلسلة لـــ DSP باستخدام مفتاح شبكة. وفي هذا السيناريو، يمكن تحقيق تجفير من نقطة إلى نقطة بواسطة مستعمل DUA لتجفير طلب DAP لـــ 'DSA 'b ويمكن أن يحقق 'DSA 'b تجفير من نقطة إلى نقطة باستخدام DSAs عن بعد.



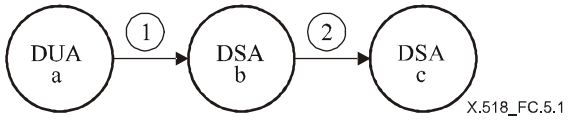
(1) DUA 'a' encrypts a DAP argument for DSA 'b'

(2) DSA 'b' decrypts the request and tries to fulfil the request; if DSA 'b' cannot fulfil the request, it uses a "net-key" to encrypt the DSA chained operation Request (including the DAP request). Chained request is sent to DSA 'c', 'd' and 'e'

5.C عمليات موزعة موقعة ومجفرة

1.5.C توقيعات من طرف إلى طرف مع تجفير من نقطة إلى نقطة

إذا أراد مستعمل 'DUA 'a أن يوقع ويجفر طلب DAP، يمكن توفير التوقيع من طرف إلى طرف ويمكن أن يحدث التجفير فقط على أساس من نقطة إلى نقطة. ويمكن أن يوقع ويجفر 'DUA 'a طلب DAP لـــ 'DSA 'b؛ ومع ذلك، لا يعرف مستعمل 'DUA 'a ما إذا كان الطلب أم لا سيسلسل في النهاية بـــ DSA ('DSA 'c) عن بعد لمعالجته. ويجفر 'DSA 'b الطلب ويتحقق من التوقيع. ثم يحاول إنجاز الطلب. وإذا قرر 'DSA 'b أن الطلب ينبغي أن يتسلسل بـــ DSA ('DSA 'c) آخر لمعالجته، فإن 'DSA 'b يجفر **ChainingArguments** DSP من أجل 'DSA 'c. ويمكن الاحتفاظ بطلب DAP الموقع الأصلي ويمرر مع **ChainingArguments** DSP المجفر.



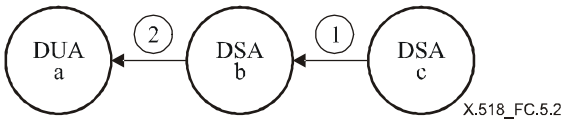
(1) DUA 'a' user signs and encrypts DAP request for DSA 'b'

(2) DSA 'b' decrypts the DAP request and verifies the signature; after trying to fulfil the request locally, DSA 'b' determines that this request needs to be chained to DSA 'c'. DSA 'b' sends the originally signed DAP Request (signed by DUA 'a' user) and generates and encrypts DSP Chaining Argument for DSA 'c'

2.5.C توقيع وتجفير من طرف إلى طرف نتيجة DAP، وتوقيع وتجفير من نقطة إلى نقطة DSP

إذا رغب مستعمل 'DUA 'a في استقبال نتائج موقعة ومجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **SIGNED-AND-ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في **ChainingArguments** ليعكس **DIRQOP** في DAP Bind. وينبغي على DSA عن بعد أن يكون له قدرة تشكيل لارسال عمليات متسلسلة موقعة ومجفرة. وفي هذا النموذج، يمكن أن ينجز نظام

'c' DSA الطلب ويولد ويؤدى تجفير من طرف إلى طرف لنتيجة DAP (لمستعمل 'a' DUA) وتجفير من نقطة إلى نقطة على **ChainingResults** DSP. ويمكنه أن يؤدى 'c' DSA بسبب أنه يعرف من هو مستعمل 'a' DUA المقصود. ويمكن تحقيق توقيع وتجفير من نقطة إلى نقطة على **ChainingResults** DSP بواسطة توقيع وتجفير 'c' DSA لـــ **ChainingResults** DSP من أجل 'b' DSA. ويمكن لـــ 'b' DSA أن يجفر ويتحقق من توقيع 'c' DSA لـــ **ChainingResults** DSP الموقع ويحيل نتيجة DAP الموقعة والمجفرة إلى مستعمل 'a' DUA. ويكون 'a' DUA مسؤولاً عن التجفير والتحقق من توقيع نتيجة DAP التي يستقبلها من 'c' DSA عبر 'b' DSA.

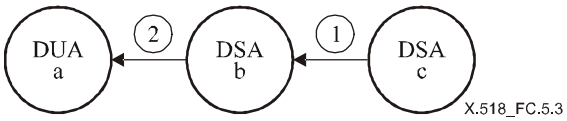


(1) DSA 'c' signs and encrypts DSP Chained Result for DSA 'b'; this includes DAP Results that are signed and encrypted for the DUA 'a' user

(2) DSA 'b' decrypts the DSP Chained Result from DSA 'c' and forwards the signed and encrypted DAP Result for DUA 'a'

3.5.C توقيع وتجفير من طرف إلى طرف DAP، وتجفير من نقطة إلى نقطة DSP ونتيجة DAP

إذا رغب مستعمل 'a' DUA في استقبال نتائج موقعة ومجفرة من الدليل، ينبغي ضبط مجال **SecurityParameters.ProtectionRequest** على **SIGNED-AND-ENCRYPTED** أو إذا لم يكن هذا المجال محيناً، يضبط مجال **SecurityParameters.ProtectionRequest** في **ChainingArguments** ليعكس **DIRQOP** في DAP Bind. وينبغي على DSA ('c' DSA) عن بعد أن يكون له قدرة تشكيل لارسال عمليات متسلسلة موقعة ومجفرة. وفي هذا النموذج، يمكن أن ينجز نظام 'c' DSA الطلب ويولد نتيجة DAP موقعة ويوقع ويجفر نتيجة DAP و**ChainingResults** DSP ويمكن لـــ 'b' DSA أن يجفر ويتحقق من توقيع 'c' DSA على **ChainingResults** DSP ويمكن لـــ 'a' DUA أن يعيد تجفير نتيجة DAP (بواسطة 'c' DSA) لمستعمل 'a' DUA. ويكون 'a' DUA مسؤولاً عن تجفير نتيجة DAP المستقبلة من 'b' DSA والتحقق من توقيع نتيجة DAP التي يستقبلها من 'c' DSA عبر 'b' DSA.



(1) DSA 'c' signs and encrypts DSP Chained Result for DSA 'b'; this includes DAP Results

(2) DSA 'b' decrypts the DSP Chained Result from DSA 'c' (and the DAP Result received in the DSP Chained Result) and forwards the signed DAP Result to DUA 'a'

الملحق D
مواصفة أنماط إسناد تراتبي وتشغيلي تراتبي غير محدد

(يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

يشمل هذا الملحق تعاريف أصناف شيء لمعلومات .ASN.1 الواردة في مواصفة الدليل هذا في شكل وحدة .ASN.1 **HierarchicalOperationalBindings**.

```
HierarchicalOperationalBindings
        {joint-iso-itu-t ds(5) module(1) hierarchicalOperationalBindings(20) 5}
DEFINITIONS ::=
BEGIN

-- EXPORTS All --
-- تصدر الأنماط والقيم المعرفة في هذه الوحدة لاستخدامها في وحدات ASN.1 الواردة في مواصفات الدليل، وللاستخدام
-- في التطبيقات الأخرى التي تستخدم للنفاذ إلى خدمات الدليل. ويمكن أن تستخدمها تطبيقات أخرى لأغراضها الخاصة،
-- ولكن لا يقيد هذا التمديدات والتعديلات المطلوبة للحفاظ على خدمة الدليل أو تحسينها.

IMPORTS

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

        directoryOperationalBindingTypes, directoryOSIProtocols, distributedOperations,
        informationFramework, opBindingManagement
            FROM UsefulDefinitions {joint-iso-itu-t ds(5) module(1) usefulDefinitions(0) 5}

        Attribute, DistinguishedName, RelativeDistinguishedName
            FROM InformationFramework informationFramework

        OPERATIONAL-BINDING
            FROM OperationalBindingManagement opBindingManagement


-- from ITU-T Rec. X.518 | ISO/IEC 9594-4

        MasterAndShadowAccessPoints
            FROM DistributedOperations distributedOperations


-- from ITU-T Rec. X.519 | ISO/IEC 9594-5

        directorySystemAC
            FROM DirectoryOSIProtocols directoryOSIProtocols

        id-op-binding-hierarchical, id-op-binding-non-specific-hierarchical
            FROM DirectoryOperationalBindingTypes directoryOperationalBindingTypes                ;

-- types --

HierarchicalAgreement ::= SEQUENCE {
        rdn                 [0]  RelativeDistinguishedName,
        immediateSuperior   [1]  DistinguishedName }
```

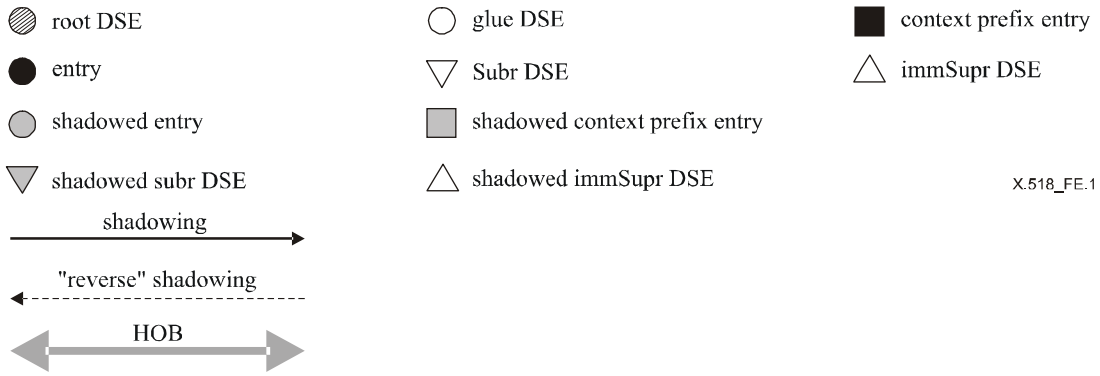
```
SuperiorToSubordinate ::= SEQUENCE {
        contextPrefixInfo           [0]  DITcontext,
        entryInfo                   [1]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        immediateSuperiorInfo       [2]  SET SIZE (1..MAX) OF Attribute OPTIONAL }

DITcontext ::= SEQUENCE OF Vertex

Vertex ::= SEQUENCE {
        rdn             [0]  RelativeDistinguishedName,
        admPointInfo    [1]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        subentries      [2]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL,
        accessPoints    [3]  MasterAndShadowAccessPoints OPTIONAL }

SubentryInfo ::= SEQUENCE {
        rdn  [0]  RelativeDistinguishedName,
        info [1]  SET OF Attribute }

SubordinateToSuperior ::= SEQUENCE {
        accessPoints [0]  MasterAndShadowAccessPoints OPTIONAL,
        alias        [1]  BOOLEAN DEFAULT FALSE,
        entryInfo    [2]  SET SIZE (1..MAX) OF Attribute OPTIONAL,
        subentries   [3]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL }

SuperiorToSubordinateModification ::= SuperiorToSubordinate (
        WITH COMPONENTS { ..., entryInfo ABSENT})

NonSpecificHierarchicalAgreement ::= SEQUENCE {
        immediateSuperior     [1]  DistinguishedName }

NHOBSuperiorToSubordinate ::= SuperiorToSubordinate (
        WITH COMPONENTS { ..., entryInfo ABSENT})

NHOBSubordinateToSuperior ::= SEQUENCE {
        accessPoints [0]  MasterAndShadowAccessPoints OPTIONAL,
        subentries   [3]  SET SIZE (1..MAX) OF SubentryInfo OPTIONAL }

-- operational binding information objects --

hierarchicalOperationalBinding OPERATIONAL-BINDING ::= {
        AGREEMENT       HierarchicalAgreement
        APPLICATION CONTEXTS {
                        {directorySystemAC} }
        ASYMMETRIC
            ROLE-A {        -- superior DSA
                ESTABLISHMENT-INITIATOR     TRUE
                ESTABLISHMENT-PARAMETER     SuperiorToSubordinate
                MODIFICATION-INITIATOR      TRUE
                MODIFICATION-PARAMETER      SuperiorToSubordinateModification
                TERMINATION-INITIATOR       TRUE }
            ROLE-B {        -- subordinate DSA
                ESTABLISHMENT-INITIATOR     TRUE
                ESTABLISHMENT-PARAMETER     SubordinateToSuperior
                MODIFICATION-INITIATOR      TRUE
                MODIFICATION-PARAMETER      SubordinateToSuperior
                TERMINATION-INITIATOR       TRUE }
        ID              id-op-binding-hierarchical }

nonSpecificHierarchicalOperationalBinding OPERATIONAL-BINDING ::= {
        AGREEMENT       NonSpecificHierarchicalAgreement
        APPLICATION CONTEXTS {
                        { directorySystemAC } }
        ASYMMETRIC
            ROLE-A {        -- superior DSA
                ESTABLISHMENT-PARAMETER     NHOBSuperiorToSubordinate
                MODIFICATION-INITIATOR      TRUE
                MODIFICATION-PARAMETER      NHOBSuperiorToSubordinate
                TERMINATION-INITIATOR       TRUE }
            ROLE-B {        -- subordinate DSA
                ESTABLISHMENT-INITIATOR     TRUE
                ESTABLISHMENT-PARAMETER     NHOBSubordinateToSuperior
```

```
        MODIFICATION-INITIATOR          TRUE
        MODIFICATION-PARAMETER          NHOBSubordinateToSuperior
        TERMINATION-INITIATOR           TRUE }
    ID              id-op-binding-non-specific-hierarchical }

END -- HierarchicalOperationalBindings
```

الملحق E
مثال صيانة معرفة

(لا يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

يوضح هذا الملحق صيانة معرفة، كما عرفت في القسم 23، بمثال بسيط. وفي الشكل E.1، تستخدم الرموز التالية لتوضيح شجرات معلومات DSA لعدد خمسة DSAs.

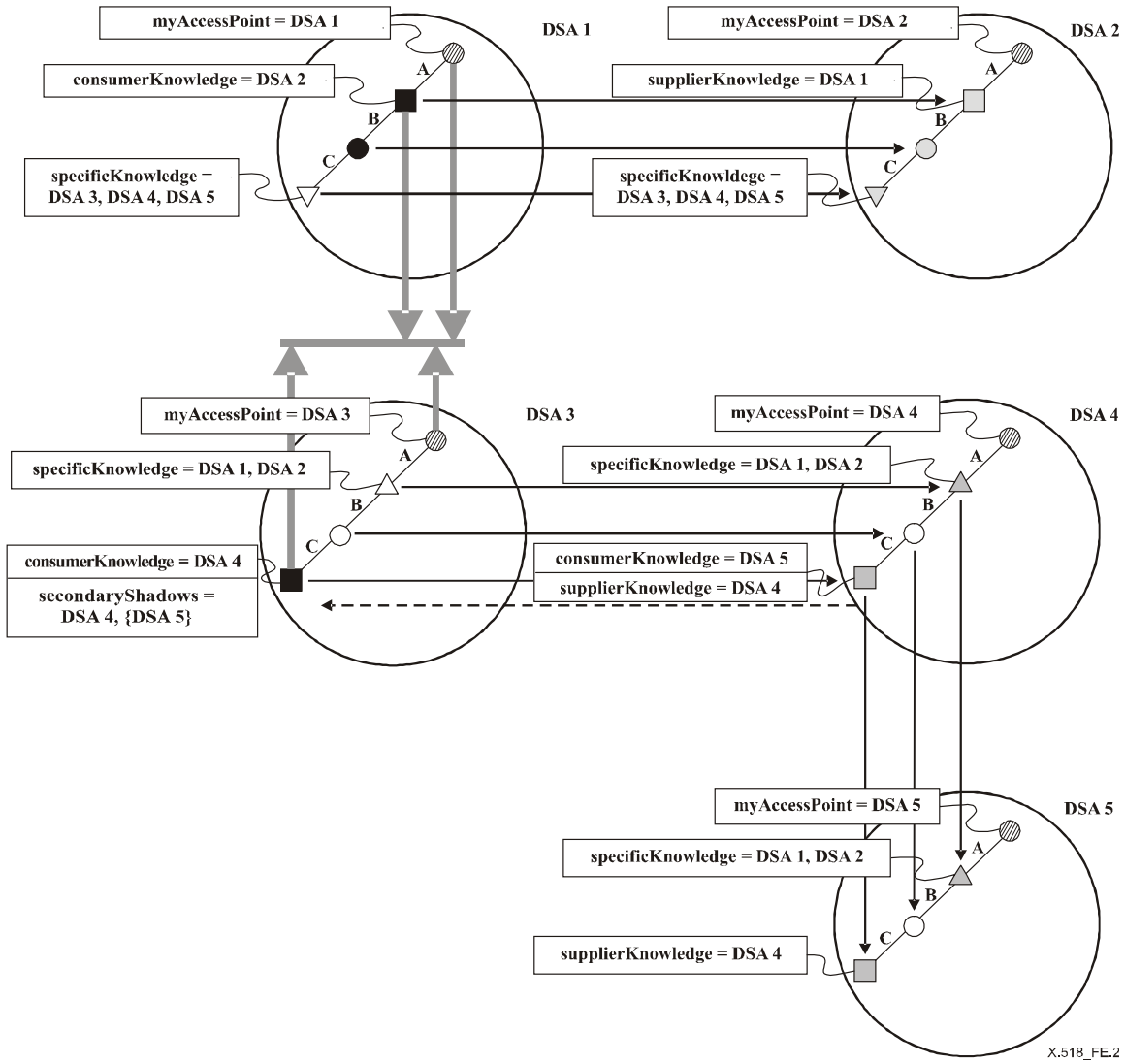


الشكل E.1 – الرموز المستخدمة لتوضيح شجرات معلومات DSA

في الشكل E.2، يكون DSA 1 هو الرئيسي لسياق تسمية {A}، يتألف من مدخلين {A} و{A,B}. ويحتفظ DSA 1 بمرجع تابع لسياق تسمية {A,B,C} الذي تجرى صيانته عبر HOB مع DSA 3. ويكون DSA 1 مورد ظل لـــ DSA 2، يزوده بنسخ من معلومات المستعمل لسياق تسمية {A} والمرجع التابع لسياق تسمية {A,B,C} الذي يعرف نقاط نفاذ DSA 3 وDSA 4 وDSA 5، ويكون الأول هو الرئيسي لسياق تسمية تابع.

يكون DSA 3 هو الرئيسي لسياق تسمية {A,B,C}. وبالإضافة إلى الاحتفاظ بمدخل وحيد {A,B,C} لسياق تسمية، يحتفظ DSA 3 بمرجع أعلى مباشر لسياق تسمية {A} الذي تجرى صيانته عبر HOB مع DSA 1. ويكون DSA 3 هو مورد ظل إلى DSA 4، يزوده بنسخ من معلومات المستعمل لسياق تسمية {A, B, C} والمرجع الأعلى المباشر لسياق تسمية {A} الذي يعرف نقاط نفاذ DSA 1 وDSA 2، ويكون الأول هو الرئيسي لسياق تسمية أعلى. ويكون DSA 4 (الثانوي) مورد ظل إلى DSA 5، يوفر له نسخة من المعلومات التي يستقبلها من DSA 3.

يوضح الشكل E.2 النعوت التشغيلية لـــ DSA المستخدمة لتقديم المعرفة وصيانتها.



الشكل 2.E – مثال صيانة معرفة

يستخدم DSA 1 قيمة نعته **myAccessPoint** (متصاحبة مع جذر DSE) والقيم القابلة للاستخدام عامة لـــ **consumerKnowledge** (متصاحبة مع نعت {A} لسابقة سياق لتشكل قيمة لنمط **MasterAndShadowAccessPoints** لاستخدامها في تفاعلات HOB مع DSA 3. ويستخدم DSA 3، بدوره، قيمة نعته **myAccessPoint** (متصاحبة مع جذر DSE) والقيم القابلة للاستخدام عامة لـــ **consumerKnowledge** و**secondaryShadows** (كلاهما متصاحب مع نعت {A,B,C} لسابقة سياق لتشكل قيمة من نمط **MasterAndShadowAccessPoints** لاستخدامها في تفاعلات HOB مع DSA 1. ويقوم DSA اثنان، معا، باستخدام DOP، بصيانة مرجع تابع يحتفظ به DSA 1 ومرجع أعلى مباشر يحتفظ به DSA 3. ويقوم المرجع التابع لـــ DSA 1، المعبر عنه بواسطة نعت **specificKnowledge** متصاحب مع DSA عند {A,B,C}، على أساس قيمة **MasterAndShadowAccessPoints** التي تستقبلها من DSA 3؛ ويكون المرجع الأعلى المباشر لـــ DSA 3 المعبر عنه بواسطة نعت **specificKnowledge** متصاحب مع DSE عند {A}، مماثل على أساس قيمة **MasterAndShadowAccessPoints** التي تستقبلها من DSA 1.

يستخدم كل من DSA 1 وDSA 2 قيمهما لـــ **myAccessPoint** في تفاعلات Shadowing Operational Binding لصيانة قيمة **consumerKnowledge** في DSA 1 (لتعريف نقطة نفاذ DSA 2) و**supplierKnowledge** في DSA 2 (لتعريف نقطة نفاذ DSA 1)، ويتصاحب كل من نعتهما مع سابقة لسياق {A}. ويقوم DSA اثنان، معاً، باستخدام DOP، بصيانة مرجع مستهلك يحتفظ به DSA 1 ومرجع أعلى يحتفظ به DSA 2.

يستقبل DSA 2 نسخة من نعت **specificKnowledge** متصاحب مع سابقة سياق {A, B, C} من DSA 1 في تفاعلات DISP مع DSA 1. وتقوم هذه التفاعلات بصيانة مرجع أعلى لـــ DSA 2 لسابقة سياق {A, B, C}.

يقوم DSA 3 وDSA 4 (وبالمثل DSA 4 وDSA 5) بصيانة مرجعي مستهلك ومورد على التوالي، بطريقة تماثلية للتفاعل بين DSA 1 وDSA 2.

يستقبل DSA 4 نسخة من نعت **specificKnowledge** متصاحب مع سابقة سياق { A4} من DSA 3 في تفاعلات DISP مع DSA 3. وتقوم هذه التفاعلات بصيانة مرجع أعلى مباشر لـــ DSA 4 لسابقة سياق { A}.

ينقل DSA 4 إلى DSA 3 أي تغييرات في **myAccessPoint** ونعت **consumerKnowledge** (ونعت **secondaryShadows**، الذي هو معدوم في هذا المثال) باستخدام تعديل عملية إسناد تشغيلي لـــ DOP. ويورد DSA 4 إلى DSA 3 قيمة **SupplierAndConsumers**، المحتوية على تلك القيم فقط لنعت **consumerKnowledge** الذي يعرف نقاط نفاذ DSAs الذين لهم ظلال قابلة للاستخدام عامة؛ وتكون قيم نعت **secondaryShadows** التي وردها DSA 4، إن وجد أي منها، حسب التصميم، قابلة للاستخدامة عامة. (وفي هذا المثال، يفترض أن DSA 5 يحتفظ بنسخة قابلة للاستخدام عامة لسابقة سياق {A,B,C}). ويستخدم DSA 3 هذه المعلومات لصيانة قيمة نعت **secondaryShadows** المتصاحبة مع سابقة سياق {A,B,C}. ويستخدم هذا النعت، كما ورد أعلاه، في تفاعلات DOP مع DSA 1 لصيانة مرجع تابع لـــ DSA 1 بسابقة سياق {A,B,C}.

ويقوم DSA 5 بصيانة مرجع أعلى مباشر لسابقة سياق {A} باستخدام تفاعلات DISP مع DSA 4 بطريقة تماثلية للتفاعل بين DSA 3 وDSA 4.

الملحق F

التعديلات والتصويبات

(لا يشكل هذا الملحق جزءاً متكاملاً من هذه التوصية/المعيار الدولي)

تشمل الطبعة هذه من مواصفة الدليل مسودة التعديلات التالية على الطبعة السابقة التي تم التصويت عليها ووافقت عليها اللجنة الكهرتقنية الدولية التابعة للمنظمة الدولية للتوحيد القياسي:

- التعديل 1 على تمديدات لدعم النتائج المتصفحة على بروتوكول نظام الدليل؛
- التعديل 3 على التراصف الأقصي بين التوصية X.500 وLDAP.

تشمل الطبعة هذه من مواصفة الدليل تصويبات تقنية تصحح تقرير العيوب التالية: 307.

سلاسل التوصيات الصادرة عن قطاع تقييس الاتصالات

السلسلة A	تنظيم العمل في قطاع تقييس الاتصالات
السلسلة D	المبادئ العامة للتعريفة
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 وتعدد الوسائط
السلسلة I	الشبكة الرقمية متكاملة الخدمات
السلسلة J	الشبكات الكبلية وإرسال إشارات البرامج الإذاعية الصوتية والتلفزيونية وإشارات أخرى متعددة الوسائط
السلسلة K	الحماية من التداخلات
السلسلة L	إنشاء الكبلات وغيرها من عناصر المنشآت الخارجية وتركيبها وحمايتها
السلسلة M	إدارة الاتصالات بما في ذلك شبكة إدارة الاتصالات (TMN) 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
السلسلة R	الإرسال البرقي
السلسلة S	التجهيزات المطرافية للخدمات البرقية
السلسلة T	المطاريف الخاصة بالخدمات التليماتية
السلسلة U	التبديل البرقي
السلسلة V	اتصالات المعطيات على الشبكة الهاتفية
السلسلة X	**شبكات المعطيات والاتصالات بين الأنظمة المفتوحة والأمن**
السلسلة Y	البنية التحتية العالمية للمعلومات وملامح بروتوكول الإنترنت وشبكات الجيل التالي
السلسلة Z	لغات البرمجة والخصائص العامة للبرمجيات في أنظمة الاتصالات

طُبعت في سويسرا
جنيف، 2006