

## DEFECT REPORT FORM

1. Defect Report Number: DR 210  
Title: poorly selected default in attribute selection in shadowing
2. Source: UK
3. Addressed to: ISO/IEC/JTC1/SC6/WG7 and ITU-T Study Group VII
4. (a) WG Secretariat: UK (BSI)  
(b) ITU-T WP: WP4
5. Date Circulated by WG Secretariat:
6. Deadline for Response from Editor:
7. Defect Report Concerning:  
ISO/IEC 9594-9
8. Qualifier: (e.g. error, omission, clarification required)  
Error
9. References in Document: (e.g. page, clause/section, figure, and/or table numbers)  
Clause 9.2.2
10. Nature of Defect: (complete, concise explanation of the perceived problem)  
It is impossible to control fully the selection of attributes for shadowing because of a faulty default expressed in the second sentence of paragraph 8: In addition, when using the **exclude** specification, any attributes contained in an entry which are not explicitly excluded are implicitly included .  
Note that in the attribute selection part of an agreement, there is a series of elements, each of which specifies an object class (optional) and one of:
  - ☒ all-user-attributes
  - ☒ a set of attributes to be included
  - ☒ a set of attributes to be excludedIf the object class is omitted, the element applies to all attributes.  
As a result of this definition, the rule requiring attributes not explicitly excluded to be implicitly included is not only completely unnecessary (to get this effect, it is only necessary to include an element {,all-user-attributes} ), but is also destructive.<sup>1</sup>.

---

<sup>1</sup> DC s comments: I initially did not agree that the rule is completely unnecessary. If you are

Supposing, for example, you wanted a person entry to **include** a list L of attributes (and its subtypes). Suppose you also wanted an object class X to **exclude** an attribute, say "mistresses-telephone-number", which was a subtype of an attribute in L. The act of specifying an exclusion causes *all* attributes now to be included, which is clearly not what is wanted. You just wanted to excise that particular attribute, without increasing the attributes to be included, and you cannot do it.

The rule also has bizarre results. Suppose, you have two elements {person,include(description)} and {person,exclude(description)}. The combined effect of this would be to include all attributes for person entries.

In any case, the sentence doesn't explicitly say whether it applies to user attributes or all attributes. The implication is the former<sup>2</sup>. Even here, there is a confusion, since an attribute which is an operational attribute for one DSA design may not be an operational attribute for another DSA. (There are more operational attributes than are

---

exclusions apply. The only sensible answer to this, is the complete set of attributes for either the entry, (or the object class if that is present). What you are proposing is that a new rule be added i.e. the set is initially empty, and you have to explicitly include all or some attributes before the exclusions can be used.

I have had several iterations of my reply, initially disagreeing with you, but finally ending up agreeing that something needs to be done. The root cause, in my opinion, is that the attribute selection clause is schizophrenic about the initial set of attributes that are present, before attribute selection takes place. On the one hand, (exclusions) the standard assumes that the initial set is the full set of attributes in the entry, whilst on the other hand (inclusions and all attributes) the standard assumes that the initial set is empty. Now this only leads to confusion, as you have pointed out, when we have a combination of excludes and includes in attribute selection. Are we including and excluding from an empty set or from the full set of attributes. It seems to me that we need to have just one model to work with, i.e. that we initially shadow no attributes, unless we add to the set, or that we shadow all attributes, unless we take from the set.

Given that there is an allAttributes in the ASN.1 that has the semantics of include everything, then we can safely start with an empty set. We cannot safely start with a full set of attributes, because we don't have an ASN.1 construct to remove everything. Therefore I agree with you that the wording needs to be changed, to say that the attributes to be shadowed is initially an empty set, and that the agreement must explicitly specify what attributes are to be shadowed. The default of course, if attributeSelection is empty, is the full set of attributes, because the ASN.1 says so. But if attribute selection is not empty, then it starts from an empty set of attributes, and adds to this through the ASN.1. The changes needed to the standard are along the following lines:

0. section 9.2.2, under principles, add a new point a) and renumber a) to b) and b) to c)
  - a) the only attributes initially included in the selection are the operational attributes specified above (i.e. access controls, timestamps and knowledge). An empty ClassAttributeSelection by default adds all user attributes to the selection, otherwise attributes have to be specifically included.

Secondly we need to delete the sentence you identified in the third paragraph under the ASN.1, i.e. In addition, when using the exclude...

<sup>2</sup> DC's comment: Agreed

defined in the standards). This is problematic, since the information contained in attribute selection is not only used for selection purposes at the supplier end, but is also used at the consumer end to help determine whether or not an enquiry can be satisfied.<sup>3</sup>

11. Solution Proposed by the Source: (optional)

Revise the clause as follows:

Amend the 5<sup>th</sup> paragraph to read:

The principles of attribute selection are:

- ) The only attributes initially included in the selection are the operational attributes specified above (i.e. access controls, timestamps and knowledge); these attributes cannot be excluded. An empty ClassAttributeSelection by default adds all user attributes to the selection, otherwise attributes have to be specifically included.
- a) The selection takes place within the shadow supplier DSA, in accordance with AttributeSelection at the time of shadowing. There are no actions whatever that are applied by the shadow consumer DSA (although the selection information may be used to determine whether or not a particular operation can be handled by the consumer DSA).
- b) Attributes that are to be selected for shadowing, SDSE by SDSE, can be selected on the basis of the class of the entries and/or subentries being shadowed, or for generic use within all shadowed entries.

In Paragraph 8,

- 0. Amend the first sentence to read: All user attributes are implicitly included\_
- 1. Remove the second sentence completely: In addition \_ .

12 Editor's Response:

---

<sup>3</sup> DC s comments: This is irrelevant in my opinion, since the attribute selection ONLY has meaning to the supplier DSA. However, it does mean that in a cascading sequence of secondary shadows, containing DSAs from different suppliers, different agreements may be needed between the supplier/consumer pairs. But this will be the case anyway.

AEH responds: it isn't the case that the attribute selection only has meaning to the supplier DSA, since it is used to determine whether an enquiry can be resolved locally; for example if an attribute X is included, an enquiry for X can legitimately be handled locally, because all the relevant information is known to be present