



1

2 SAML credentials-collector use-cases 3 and requirements

4 Working draft 01, 30 September 2003

5 Document identifier: oasis-sstc-v2.0-credentials-collector-use-cases-wd-01

6 Location: http://www.oasis-open.org/committees/documents.php?wg_abbrev=security

7 Send comments to: security-services-comment@lists.oasis-open.org

8 Editors:

9 Tim Moses, Entrust

10 Contributors:

11 Jeff Hodges, Sun Microsystems

12 Hal Lockhart, BEA

13 Abstract:

14 This working draft defines use-cases and requirements for a protocol by which system
15 entities can authenticate to a credentials-collector with the help of an authentication
16 authority.

17 Status:

18 This version of the specification is a working draft of the committee. As such, it is expected
19 to change prior to adoption as an OASIS standard.

20 If you are on the security-services@lists.oasis-open.org list for committee members, send
21 comments there. If you are not on that list, subscribe to the [security-services-
comment@lists.oasis-open.org](mailto:security-services-
22 comment@lists.oasis-open.org) list and send comments there. To subscribe, send an email
23 message to security-services-comment-request@lists.oasis-open.org with the word
24 "subscribe" as the body of the message.

25

26 Copyright (C) OASIS Open 2003 All Rights Reserved.

27	Table of contents	
28	1. Introduction	3
29	2. Use-cases	3
30	2.1. Use-case 1: Authn authority as credentials-collector	3
31	2.2. Use-case 1: Credentials-collector as intermediary	4
32	10. Requirements	7
33	11. References	7
34	Appendix A. Notices	8
35		
36		

37 1. Introduction

38 This document describes use-cases and requirements for a protocol by which a system entity can
 39 authenticate bi-laterally with a credentials-collector, using authentication services provided by an
 40 authentication authority.

41 The protocol is intended to support a broad range of authentication mechanisms, including those
 42 that result in a secret shared between the system entity and the credentials collector for protecting
 43 a subsequent session.

44 2. Use-cases

45 2.1. Use-case 1: Authn authority as credentials-collector

46 Use-case 1 is shown in Figure 1.

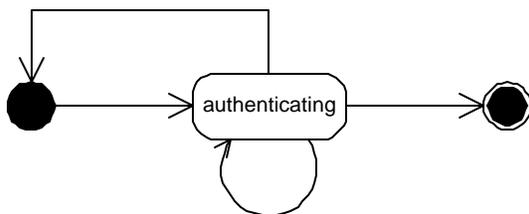


47

48 **Figure 1 - Use-case 1**

49 In this use-case the functions of the credentials-collector are performed by the authentication
 50 authority.

51 The state transition diagram is shown in Figure 2.

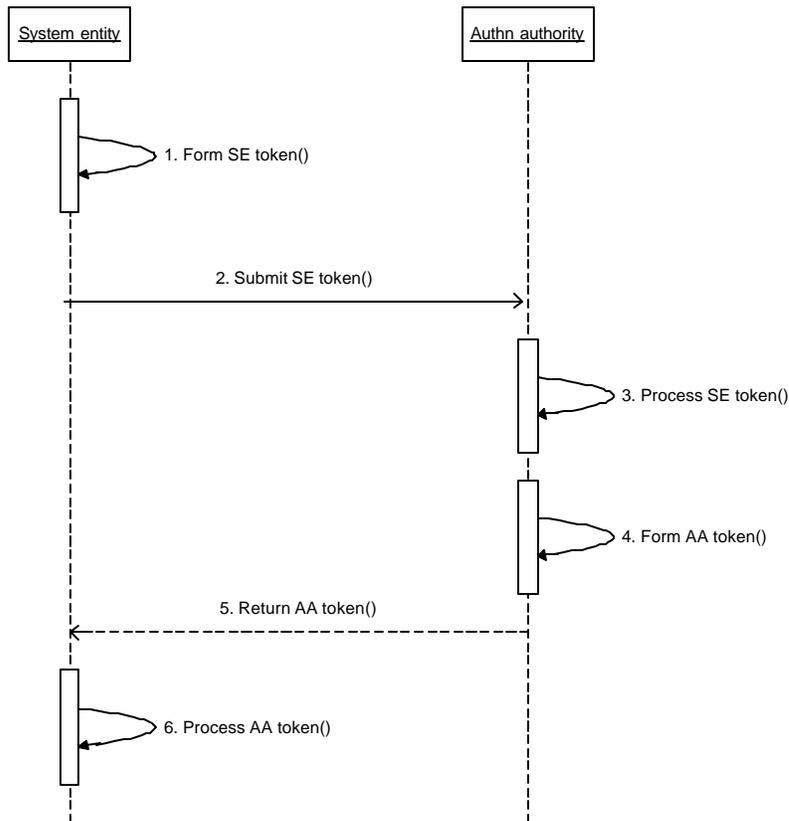


52

53 **Figure 2 - State transition diagram**

54 In this diagram, the initial state corresponds to the unauthenticated state and the final state
 55 corresponds to the authenticated state.

56 The sequence of activities in the “authenticating” state is shown in Figure 3.



57

58 **Figure 3 - Use-case 1 “authenticating” sequence**

59 1. System entity forms an authentication token (this step is optional on the first occurrence).

60 2. System entity sends the token to authentication authority.

61 3. Authentication authority processes the token and decides whether system entity is
62 authenticated or not.

63 4. If system entity is not authenticated, then authentication authority forms a token. Optionally, if
64 system entity is authenticated, then the token is an authentication assertion.

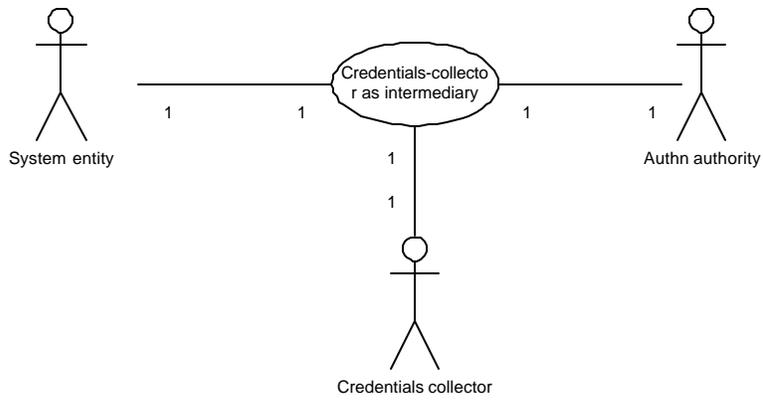
65 5. Authentication authority returns the token to systeme entity.

66 6. System entity processes the token.

67 **[SASLib]** specifies a solution for this use-case, based on the Simple Authentication and Security
68 Layer.

69 **2.2. Use-case 1: Credentials-collector as intermediary**

70 Use-case 2 is shown in Figure 4.

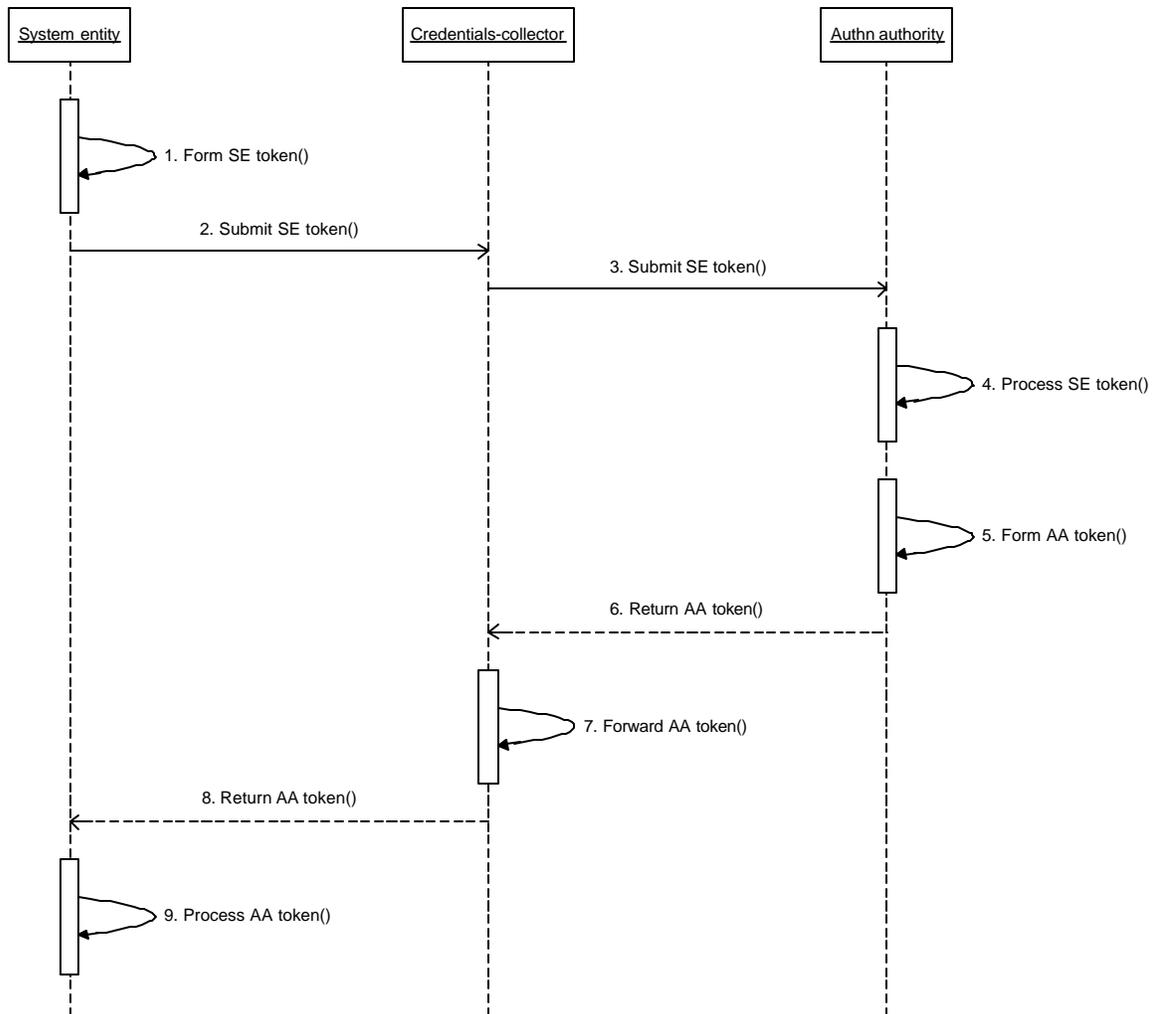


71

72 **Figure 4 - Use-case 2**

73 The state transition diagram is identical to the one shown in Figure 2.

74 The sequence of activities in the “authenticating” state is shown in Figure 5.



75

76 **Figure 5 - Use-case 2 sequence**

77 1. System entity forms an authentication token (this step is optional on the first occurrence).

78 2. System entity sends the token to credentials-collector.

79 3. Credentials-collector forwards the token to authentication authority.

80 4. Authentication authority processes the token and decides whether system entity is
81 authenticated or not.

82 5. If system entity is not authenticated, then authentication authority forms a token. If system
83 entity is authenticated, then the token is an authentication assertion.

84 6. Authentication authority returns the token.

- 85 7. If the authentication is complete, then credentials-collector extracts the authentication
86 assertion.
- 87 8. If authentication is not complete, then credentials-collector forwards the token to system entity.
- 88 9. System entity processes the token.
- 89 In the case where the authentication mechanism results in a secret shared between the
90 authentication authority and the system entity, the resulting secret is passed to the credentials-
91 collector in the subject-confirmation element of the authentication assertion.
- 92 **[SASLib]** does not currently address this use case. However, with straightforward modification, it
93 could be made to address it.

94 10. Requirements

- 95 In the case where the shared secret is a secret-key, its confidentiality must be protected in step 6 of
96 Section 2.2. This may be accomplished by means of a secure session between the authentication
97 authority and the credentials-collector or by encrypting the shared-secret for the credentials-
98 collector. The former case may not be suitable if the credentials-collector forwards the assertion.
99 In this case, the credentials-collector could remove the shared secret from the assertion. But, this
100 may disrupt any integrity protection applied to the assertion by the authentication authority.
- 101 For these reasons, encrypting the shared-secret for the credentials-collector may be the most
102 appropriate solution.

103 11. References

- 104 **[SASLib]** Liberty SASL-based SOAP Authentication Specification, Version: 1.0-03, Jeff Hodges,
105 located at: <http://www.projectliberty.org/specs/draft-lib-arch-soap-authn-v1.0-03.pdf>

106 **Appendix A. Notices**

107 OASIS takes no position regarding the validity or scope of any intellectual property or other rights
108 that might be claimed to pertain to the implementation or use of the technology described in this
109 document or the extent to which any license under such rights might or might not be available;
110 neither does it represent that it has made any effort to identify any such rights. Information on
111 OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS
112 website. Copies of claims of rights made available for publication and any assurances of licenses to
113 be made available, or the result of an attempt made to obtain a general license or permission for
114 the use of such proprietary rights by implementors or users of this specification, can be obtained
115 from the OASIS Executive Director.

116 OASIS has been notified of intellectual property rights claimed in regard to some or all of the
117 contents of this specification. For more information consult the online list of claimed rights.

118 OASIS invites any interested party to bring to its attention any copyrights, patents or patent
119 applications, or other proprietary rights which may cover technology that may be required to
120 implement this specification. Please address the information to the OASIS Executive Director.

121 Copyright (C) OASIS Open 2003. All Rights Reserved.

122 This document and translations of it may be copied and furnished to others, and derivative works
123 that comment on or otherwise explain it or assist in its implementation may be prepared, copied,
124 published and distributed, in whole or in part, without restriction of any kind, provided that the above
125 copyright notice and this paragraph are included on all such copies and derivative works. However,
126 this document itself may not be modified in any way, such as by removing the copyright notice or
127 references to OASIS, except as needed for the purpose of developing OASIS specifications, in
128 which case the procedures for copyrights defined in the OASIS Intellectual Property Rights
129 document must be followed, or as required to translate it into languages other than English.

130 The limited permissions granted above are perpetual and will not be revoked by OASIS or its
131 successors or assigns.

132 This document and the information contained herein is provided on an "AS IS" basis and OASIS
133 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO
134 ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY
135 RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
136 PARTICULAR PURPOSE.