Backend Attribute Exchange Architecture and Interface Specification

Version 1.0.0
Final
May 15, 2008

Document History

Status	Release	Date	Comment	Audience
Template	0.0.0	10/24/07	Outline	AWG
Draft	0.0.1	11/5/07	Added content to section 1	Internal
Draft	0.0.2	11/7/07	Revised per internal review	AWG
Draft	0.0.3	11/9/07	Revised per AWG review	Internal
Draft	0.04	11/12/07	Revised per internal review	AWG
Draft	0.0.5	11/19/07	Revised per AWG review	Internal
Draft	0.0.6	11/19/07	Revised per internal review	AWG
Draft	0.0.7	11/26/07	Revised per internal review	Internal
Draft	0.0.8	11/28/07	Revised per internal review	AWG
Draft	0.0.9	12/5/07	Revised per AWG review	internal
Draft	0.0.10	12/11/07	Revised Per AWG	internal
Draft	0.0.11	12/12/07	Revised per Internal discussion	AWG
Draft	0.0.12	12/31/07	Revised to reflect decision to use SAML as the BAE protocol	AWG
Draft	0.0.13	1/9/08	Revised to include SAML 2.0 protocol content	Internal
Draft	0.0.14	1/11/08	Revised per internal review	Internal
Draft	0.0.15	1/14/08	Revised per internal review	Internal
Draft	0.0.16	1/16/08	Revised per internal review	AWG
Draft	0.0.17	1/18/08	Revised per internal review	Internal
Draft	0.0.18	1/23/08	Revised per internal review	Internal
Draft	0.0.18	1/23/08	Revised per internal review	Internal
Draft	0.0.19	1/25/08	Revised per internal review	Internal
Draft	0.0.20	1/30/08	Revised per AWG comments	AWG
Draft	0.1.0	1/31/08	Revised per internal review	Internal
Draft	0.1.1	2/29/08	Revised per AWG comments	AWG
Draft	0.1.2	3/5/08	Revised per internal review	AWG
Draft	0.1.3	3/10/08	Revised per internal review	AWG
Draft	0.1.4	3/14/08	Revised per internal review	AWG
Draft	0.1.5	3/14/08	Revised per internal review	AWG
Draft	0.2.0	3/14/08	Finalized for public comment	External
Draft	0.2.1	5/1/08	Revisions per public comment	Internal
Draft	0.2.2	5/2/08	Revisions per public comments and internal review	Internal
Draft	0.2.3	5/6/08	Revisions per internal review	Internal
Draft	0.2.4	5/9/08	Revisions per internal review	Internal
Draft	0.2.5	5/12/08	Revisions per internal review	Red Team
Draft	0.2.6	5/14/08	Revisions per red team review	Internal
Draft	0.3.0	5/14/08	Final revisions per internal review	FICC
Final	1.0.0	5/15/08	Delivery to GSA	

Editors

Tim Baldridge	Chris Brown	Treb Farrales
Larry Fobian	Bradley Hiddemen	Steve Lazerowich
Chris Louden	Terry McBride	Jonathan Rich
Dave Silver	Dave Simonnetti	Carl Weber

Table of Contents

Figures

Tables

1 INTRODUCTION

1.1 Background

[FIPS 201] defines a government-wide interoperable identification credential for controlling physical access to federal facilities and logical access to federal information systems. The FIPS 201 credential, known as the Personal Identity Verification (PIV) Card, supports PIV Cardholder authentication using information securely stored on the PIV Card. Some PIV Cardholder information is available on-card through PIV Card external physical topology (i.e., card surface) and PIV Card internal data storage (e.g. magnetic stripe, integrated circuit chip). Other PIV Cardholder information is available off-card.

A Relying Party (RP) may require PIV Cardholder information directly from an authoritative source for purposes including, but not limited to PIV Card tamper detection, access decisions, provisioning in advance of access to meetings at other agency locations, and dealing with an employee or contractor medical emergency. By obtaining PIV Cardholder information directly from an authoritative source (rather than relying on or being limited by PIV Cardholder information stored on-card), the RP gains benefits such as:

1. Enhanced detection of PIV Card tampering;
2. Enhanced access control and management; and
3. Enhanced response capabilities (e.g., first responder).

Accordingly, the federal government requires a standard mechanism for RPs to obtain PIV Cardholder information (Backend Attributes) directly from the authoritative source (Attribute Authority). The authoritative source is the PIV Card Issuing Agency (PIV Card Issuer), which is the agency that issued the PIV Card to the PIV Cardholder[1]. Access to Backend Attributes is either in real-time when immediately needed (e.g., guard suspects PIV Card tampering), or in advance of need (e.g., provisioning access to a scheduled meeting, loading a handheld device prior to field use).

The exchange of Backend Attributes between backend systems is known as “Backend Attribute Exchange” (BAE)[2].

1.2 BAE Basic Concept

BAE is a general concept pertaining to exchange of PIV Cardholder information in a secure and trusted environment between an Attribute Authority (AA) and an RP. There are two BAE models and corresponding interface specifications that can be implemented:

1. **Single PIV Cardholder BAE Model** – Security Assertion Markup Language (SAML) based exchange of Backend Attributes for one PIV Cardholder per request/response pair.
2. **Batch Processing BAE Model** – Service Provisioning Markup Language (SPML) based exchange of Backend Attributes for multiple PIV Cardholders per request/response pair.

A federal agency may use one or both BAE models, as circumstances dictate. The basic principles and objectives are the same for each BAE model. The RP obtains all requested Backend Attributes from the AA via BAE Brokers, even those Backend Attributes that may already be stored on-card. The AA is the PIV Card Issuer and authoritative source for its PIV Cardholder information. The RP initiates a Backend Attribute request. A PIV Card may or may not be present when the request is made, depending upon the use case. Backend Attributes include but are not limited to PIV Cardholder photograph, PIV Cardholder fingerprints, PIV Cardholder emergency contact

[1] See Section 3.5.4. In the future, additional authoritative sources may be supported.

[2] BAE was previously known as “Backend Authentication”

information, PIV Cardholder security clearance level, and PIV Cardholder emergency responder capabilities. See Appendix A for a list of currently supported Backend Attributes. The RP uses returned Backend Attributes as necessary.

1.3 Objective and Audience

This document's primary objective is to define an interoperable model and interface for government-wide BAE. This document provides a high-level description of BAE business use cases, BAE business processes, the BAE architectural model, and standards-based BAE interface specifications. Some sections are normative (e.g., interface specification), while other sections are informational or recommendations (e.g., governance).

1.4 Scope

This document defines the end-to-end architectural model and interface specification for inter-agency exchange of Backend Attributes. The exchange is ultimately between RP and AA systems. Scope is limited to explaining the two BAE models and defining each model's interface specification.

BAE interface specifications are limited to defining technical interoperation between agency communications conduits called BAE Brokers (see Sections 3.4 and 3.5.1).

This document does not address BAE governance, trust, and privacy matters (see Section 3.3).

This document does not supersede or contradict any existing National Institute of Standards and Technology (NIST) publication, and should be used in conjunction with existing policies and procedures – particularly [NIST 800-47] and its guidelines for planning, establishing, maintaining, and terminating interconnections between information technology (IT) systems that are owned and operated by different organizations.

1.5 Authority

The Homeland Security Presidential Directive 12 (HSPD-12) Architecture Working Group (AWG) developed this document on behalf of the Office of Governmentwide Policy and the HSPD-12 Executive Steering Committee in furtherance of their charter to implement HSPD-12 from a "national" perspective.

1.6 Methodology

The BAE architecture and interface specifications presented herein are the result of the following methodology:

1. **Identification of Use Cases and Business Processes –** The HSPD-12 AWG identified and analyzed various use cases and business processes that BAE needs to address. While doing so, the HSPD-12 AWG developed business requirements and assumptions.
2. **Identification of Candidate Architecture Models** – The HSPD-12 AWG identified and analyzed various conceptual architecture models that could address the use cases. The HSPD-12 AWG assessed the pros and cons of each model, including but not limited to complexity, practicality, and governance. The HSPD-12 AWG added and revised candidates through sharing of experiences and brainstorming. During this step, the HSPD-12 AWG developed technical requirements and assumptions.
3. **Selection of BAE Architecture Model** – The HSPD-12 AWG identified the optimal architecture model to support the use cases. The decision addressed technical, operational, and business considerations.
4. **Identification of Candidate Protocol Standards** – The HSPD-12 AWG identified and analyzed various protocols that could serve as the basis for each BAE model's interface

specification. The HSPD-12 AWG focused on industry standard protocols. The HSPD-12 AWG assessed the pros and cons of each protocol.

5. **Selection of BAE Protocol Standards** – The HSPD-12 AWG identified the optimal protocol to serve as the basis of each BAE model's BAE interface specification. The decision addressed technical, operational, and business considerations.

1.7 Document References

The following is a list of documents that will be of interest to BAE participants. The documents provide additional insights, guidance, and requirements. Some documents may be relevant for one task only. Other documents may be relevant in many places.

This document uses the NIST convention for citing documents. The shorthand format [*Doc Reference*] indicates a document fully cited in this section. For example, [FIPS 201] refers back to this section's citation for the *FIPS 201-1, Personal Identity Verification (PIV) of Federal Employees and Contractors, NIST, March 2006* document. This convention reduces verbiage throughout the document.

[BAE Use Cases] Use Cases for Defining Backend Attribute Exchange
http://www.smart.gov/awg/

[FASC-N] Technical Implementation Guidance: Smart Card Enabled Physical Access Control Systems; Physical Access Interagency Interoperability Working Group
http://www.smart.gov/iab/documents/PACS.pdf

[FIPS 10-4] Federal Information Processing Standards Publication 10-4; Countries, Dependencies, Areas of Special Sovereignty, and Their Principal Administrative Divisions
http://www.itl.nist.gov/fipspubs/fip10-4.htm

[FIPS 201] FIPS 201-1, Personal Identity Verification (PIV) of Federal Employees and Contractors, NIST, March 2006
http://csrc.nist.gov/publications/fips/fips201-1/FIPS-201-1-chng1.pdf

[GSA USAccess] GSA HSPD-12 USAccess Program Authoritative User Data Interface Specification
Contact the Managed Service Office

[HSPD-12] Homeland Security Presidential Directive/HSPD-12, "Policy for a Common Identification Standard for Federal Employees and Contractors"; August 27, 2004
http://csrc.ncsl.nist.gov/policies/Presidential-Directive-Hspd-12.html

[NIPP] "National Infrastructure Protection Plan" Department of Homeland Security, 2006
http://www.dhs.gov/xlibrary/assets/NIPP_Plan.pdf

[NRP] “National Response Plan” Department of Homeland Security, December 2004
http://www.dhs.gov/xlibrary/assets/NRP_FullText.pdf

[NIST 800-47] Security Guide for Interconnecting Information Technology Systems
National Institute of Standards and Technology
http://csrc.nist.gov/publications/nistpubs/800-47/sp800-47.pdf

[NIST 800-52] Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations
http://csrc.nist.gov/publications/nistpubs/800-47/sp800-47.pdf

[NIST 800-87] Codes for the Identification of Federal and Federally Assisted Organizations
http://csrc.nist.gov/publications/nistpubs/800-87/sp800-87-Final.pdf

[NIST 800-95] Guide to Secure Web Services
National Institute of Standards and Technology
http://csrc.nist.gov/publications/nistpubs/800-47/sp800-47.pdf

[RFC 2119] Key words for use in RFCs to Indicate Requirement Levels
http://www.ietf.org/rfc/rfc2119.txt

[SAML2 Bindings] “Bindings for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005. Document Identifier: saml-bindings-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf

[SAML2 Conform] “Conformance Requirements for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005.
Document Identifier: saml-conformance-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-conformance-2.0-os.pdf

[SAML2 Context] “Authentication Context for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005.
Document Identifier: saml-authn-context-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf

[SAML2 Core] “Assertions and Protocol for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005.
Document Identifier: saml-core-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

[SAML2 Glossary] “Glossary for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005. Document Identifier: saml-glossary-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-glossary-2.0-os.pdf

[SAML2 Metadata] “Metadata for the OASIS Security Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005.
Document Identifier: saml-metadata-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf

[SAML2 Metadata Ext] Metadata Extension for SAML V2.0 and V1.x Query Requesters; OASIS Standard; 1 November 2007
http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-ext-query-os.pdf

[SAML2 Profiles] "Profiles for the OASIS Security Markup Language (SAML) V2.0", OASIS Standard, 15 March 2005. Document Identifier: saml-profiles-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

[SAML2 Security] "Security and Privacy Considerations for the OASIS Security Markup Language (SAML) V2.0", OASIS Standard, 15 March 2005.
Document Identifier: saml-sec-consider-2.0-os
http://docs.oasis-open.org/security/saml/v2.0/saml-sec-consider-2.0-os.pdf

[SOAP] Simple Object Access Protocol (SOAP) 1.1; W3C
http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[SPML2] "OASIS Service Provisioning Markup Language (SPML) Version 2.0", OASIS Standard, 1 April 2006. Document Identifier: pstc-spml2-os.pdf
http://www.oasis-open.org/committees/download.php/17708/pstc-spml-2.0-os.zip

[WS-Security] "Web Services Security: SOAP Message Security 1.1(WS-Security 2004)"; OASIS Standard, 1 February 2006
http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

1.8 Web Site References

Topic	Links
JPEG 2000	http://www.jpeg.org/jpeg2000
NIEM	http://www.niem.gov/ http://www.niem.gov/topicIndex.php?topic=documentation
NIST Documents	http://csrc.nist.gov/publications
SAML	http://www.oasis-open.org/home/index.php http://www.oasis-open.org/specs/index.php#samlv2.0 http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security http://www.oasis-open.org/committees/security/docs
SOAP	http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
SPML	http://www.oasis-open.org/specs/index.php#spmlv2.0 http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=provision
WS-Security	http://www.oasis-open.org/committees/workgroup.php?wg_abbrev=ws-sx http://www.oasis-open.org/committees/workgroup.php?wg_abbrev=wss http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss http://www.ibm.com/developerworks/library/specification/ws-secure/
XML	http://www.w3.org/1999/XMLSchema-instance http://www.w3.org/1999/XMLSchema
XPATH	http://www.w3.org/TR/xpath

1.9 Notation

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119].

2 BAE Business Requirements and Assumptions

2.1 BAE Business Requirements

BAE addresses the following high-level business requirements:

- **Scope of Functionality:** BAE MUST provide federal agencies with a mechanism to access the Backend Attributes from other agencies to facilitate access control decisions and help manage emergency situations, among other uses.
- **Privacy Protection**: Privacy and confidentiality of Backend Attributes MUST be protected.
- **Policy Compliant**: BAE MUST comply with applicable policy framework requirements (e.g., [NIST SP 800-95], [HSPD-12]).
- **Service Transaction Context**: BAE MUST support conditions where PIV Cardholder is present and not present.
- **Support for Smaller Agencies**: BAE MUST support use by smaller agencies. Smaller agencies SHOULD be provided the opportunity to leverage existing BAE architectural components whether provided and run by other agencies or by shared services.
- **Quality of Service:** BAE MUST be reliable, highly available, secure and auditable**.**
- **Types of Service:** BAE SHOULD provide different kinds of service to support single and batch requests.
- **Mandatory and Optional Attributes:** BAE SHOULD define a set of Backend Attributes to be exchanged between agencies as well as their acceptable values, and indicate which Backend Attributes are mandatory and which are optional.
- **Open Data Model:** BAE MUST allow the defined set of Backend Attributes to be modified over time, to support agencies needs. Agencies SHOULD be able to request Backend Attribute table modifications on a per-BAE-release basis.
- **Balanced Approach**: To facilitate government-wide BAE adoption, a proper balance SHOULD be achieved between convenience (i.e., ease of implementation, use, and maintenance) on the one hand, and security and privacy on the other.
- **Cost-effective**: BAE MUST be financially viable to implement and maintain.
- **Standards-based**: BAE SHOULD rely on existing industry standards while remaining aware of emerging standards.
- **Distributed, Brokered Trust Relationships**: BAE will be based on distributed trust domains with relationships managed by the Federal Identity Credentialing Committee (FICC).

2.2 BAE Business Assumptions

BAE makes the following high-level business assumptions:

- **Governance**: Governance for BAE will be provided to control who is allowed to participate in BAE, and to administer accreditation, provisioning, and configuration of any necessary trust relationships between participants.
- **Use of Information Received**: Agencies can use Backend Attributes in any way consistent with federal privacy and security guidelines in general, and their agency’s privacy and security requirements and guidelines in particular.
- **PIV Card Validation**: If a PIV Card is present, the RP will validate PIV Card certificates when conducting a BAE transaction.
- **Identification of PIV Cardholder**: Identification of the PIV Cardholder and the authoritative source for their Backend Attributes will be based on the PIV Card Federal Agency Smart Credential Number (FASC-N) and the organization code contained within the FASC-N.
- **Attribute Authorities**: For initial deployment, AAs are PIV Card Issuers.
    - AAs have the information necessary to support all mandatory Backend Attributes.
    - Future BAE versions will likely support other types of AAs.

3 BAE Architecture

The BAE architecture is a technical framework into which approved components integrate and technically interoperate via well-defined interface specifications.

3.1 Architecture Assumptions and Considerations

The BAE architecture assumes the following:

1. BAE deployment will be a phased approach.
2. Initially, there will be a few BAE participants (e.g., 15 or fewer). Participation will increase over time. While a small number of participants, BAE may use non-automated approaches to reduce effort or cost, and to expedite roll out.
3. Placing BAE Brokers behind a gateway to segregate the BAE Broker, which is privy to personal information, from the Internet is an important consideration but out of scope for this document.
4. For each PIV Cardholder, a single AA has primary knowledge of the PIV Cardholder, and knows all other AAs (across organizations) that contain information about the PIV Cardholder. For initial deployment, the response side of BAE processing collects all requested Backend Attributes, regardless of where located.
5. For initial deployment, BAE requests and responses pertain to Backend Attributes only. In the future, other types of requests and responses may be added.

3.2 Design Goals

The BAE technical vision derives from the following design goals:

1. **Commercial-off-the-Shelf (COTS)**: The architecture SHOULD employ COTS products wherever possible;
2. **Durable**: The architectural framework SHOULD be designed to allow for the evolution of technology, providing for easy migration as the industry evolves;
3. **Flexible**: The architectural framework SHOULD not rely on any single standard, vendor, product, or integrator;
4. **Scalable**: The solution MUST be scalable both technologically and administratively;
5. **Reliable:** The architecture MUST be very dependable, applying best practices and establishing a high level of credibility and confidence;
6. **Ease of use:** The end user experience SHOULD be as simple as possible by optimizing usability, availability, and response times;
7. **Ease of adoption:** Agency adoption MUST be optimized by mitigating technical barriers to entry;
8. **Extensible**: The architecture SHOULD readily support additional use cases and exchange of additional Backend Attributes;
9. **Seamless**: BAE participants and components SHOULD be minimally affected by future BAE architecture or BAE interface specification changes;
10. **Discovery:** Where applicable, determination of the BAE Responder (see Section 3.5.1) to send a request must be obtainable from information within the PIV Card.

3.3 Governance and Trust

The HSPD-12 AWG defers governance, trust, and privacy matters to the FICC. The FICC should address issues including, but not limited to the following:

- BAE component security controls;
- Institution of BAE auditing and accreditation requirements;

- Assessment of BAE components and agency systems participating in BAE;
- Issuance of BAE Trust Certificates;
- Dissemination of BAE metadata to BAE participants (see Sections 4.3.4 and 5.4); and
- Privacy and confidentiality including PIV Cardholder consent (i.e. end user BAE opt-in).

3.4 Conceptual BAE Architecture

Figure 3-1 depicts the conceptual BAE architecture, which supports both BAE models. Inter-agency communication and data exchange are accomplished via BAE Brokers. All communication is via request/response message pairs. A BAE Broker can be implemented at different organizational levels (e.g., Agency level, Department level). The organizational level chosen must have an associated [NIST 800-87] code for BAE Broker routing purposes.

Figure 3-1 Conceptual BAE Architecture

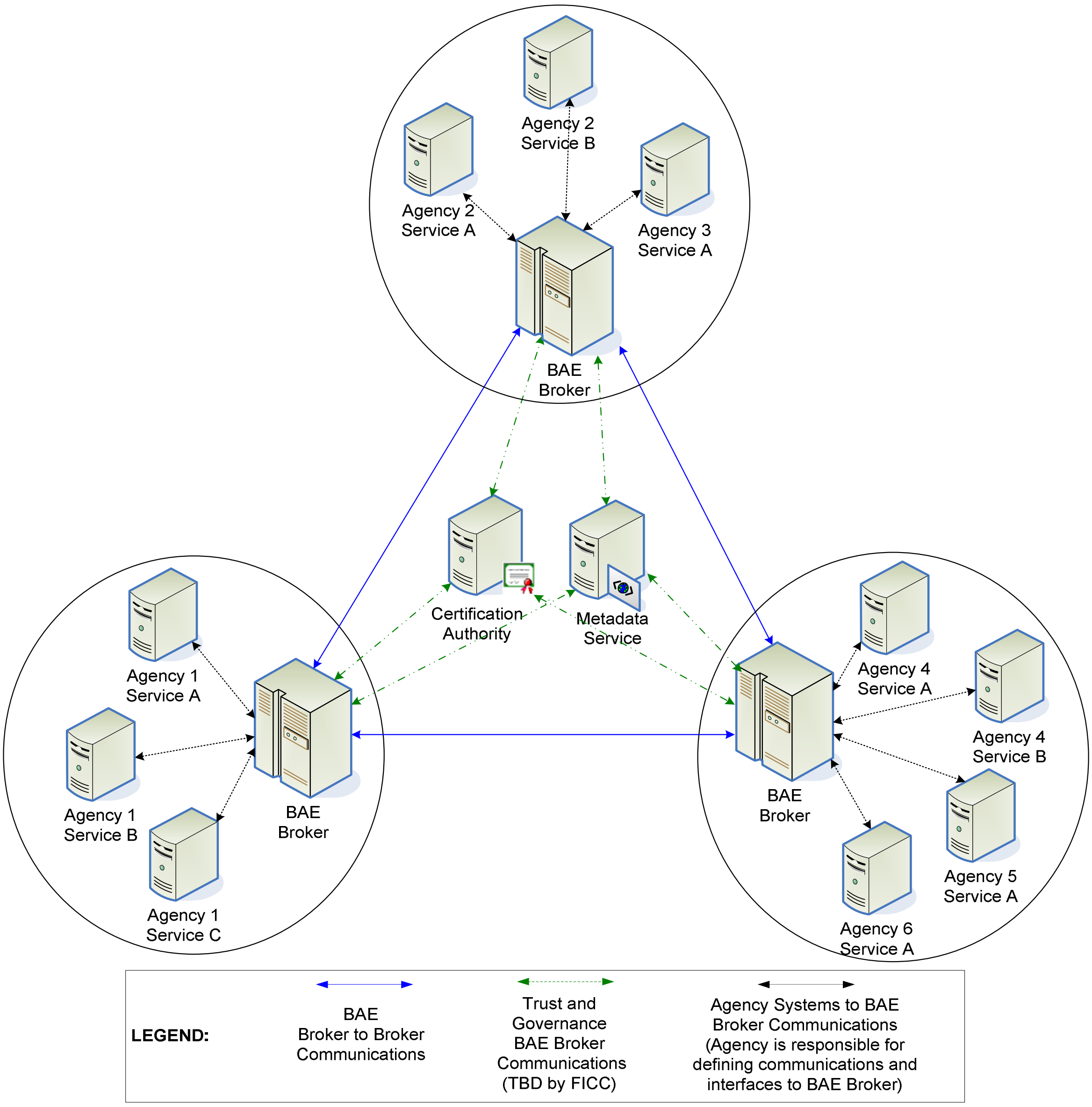


Figure 3-2 illustrates the conceptual BAE process flow, which Table 3-1 describes. Prior to live operations, BAE Brokers are configured with PKI certificates and metadata to support trusted technical interoperability. Initially, metadata exchange will likely be manual and out-of-band.

Figure 3-2 Conceptual BAE Request/Response Process Flow

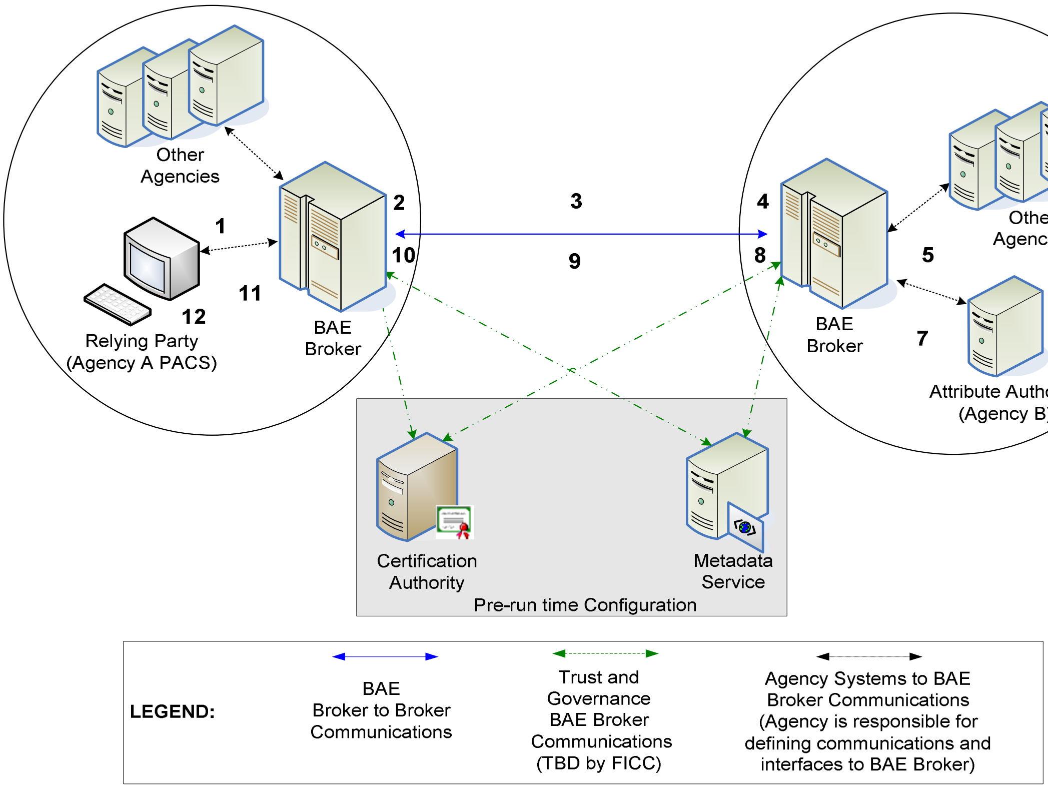

Other Agencies
1
BAE Broker
2
10
11
12
Relying Party (Agency A PACS)
3
9
4
8
BAE Broker
Other Agencies
5
6
7
Attribute Authority (Agency B)
Certification Authority
Metadata Service
Pre-run time Configuration
LEGEND:
BAE Broker to Broker Communications
Trust and Governance BAE Broker Communications (TBD by FICC)
Agency Systems to BAE Broker Communications (Agency is responsible for defining communications and interfaces to BAE Broker)

Table 3-1 Conceptual BAE Request/Response Process Flow

Step	Description
1	The RP submits a request to its BAE Broker. This interface is defined within each agency and is out of scope for this document.
2	The RP's BAE Broker processes the request as necessary.
3	The RP's BAE Broker then routes the request to the appropriate AA's BAE Broker. This interface is defined by each BAE model (See Sections 4.3 and 5.3).
4	The AA's BAE Broker processes the request as necessary.
5	The AA's BAE Broker then routes the request to the appropriate AA. This interface is defined within each agency and is out of scope for this document.
6	The AA processes the request as necessary, packaging a response as appropriate.
7	The AA sends a response to its BAE Broker. This interface is defined within each agency and is out of scope for this document.
8	The AA's BAE Broker processes the response as necessary.
9	The AA's BAE Broker then routes the response back to the requesting RP's BAE Broker. This interface is defined by each BAE model (See Sections 4.3 and 5.3).
10	The RP's BAE Broker processes the response as necessary.
11	The RP's BAE Broker then routes the response to the RP. This interface is defined within each agency and is out of scope for this document.
12	The RP processes the response as necessary.

3.5 Components

BAE includes (a) BAE Brokers that must demonstrate compliance with applicable BAE interface specifications before deployment, and (b) agency systems that communicate with each other via BAE Brokers. Components comprising the BAE architecture may or may not reside on the same physical machine. The specific implementation of components is determined by each participating organization.

3.5.1 BAE Broker

The BAE Broker is the communications conduit between RPs and AAs. The BAE Broker includes (1) an Internal BAE Service, and (2) an External BAE Service. External BAE Services exchange Backend Attributes between trusted BAE partners.

When making a request (e.g., requesting Backend Attributes), the BAE Broker is a BAE Requester. When returning a response (e.g., returning Backend Attribute values), the BAE Broker is a BAE Responder.

The External BAE Service processes transactions as necessary, including but not limited to the following:

- Message signing and signature verification;
- Message encryption and decryption; and
- Message routing.

BAE Brokers are configured with BAE metadata as necessary to facilitate trusted, secure technical interoperation and transaction processing.

3.5.1.1 External BAE Service

The External BAE Service is an inter-agency communications mechanism. External BAE Services communicate directly with each other to securely exchange BAE messages. Communication is in a request-response manner.

In the Single PIV Cardholder BAE model, the request message is the RP’s list of desired Backend Attributes, and the response message is the Backend Attribute values returned by the AA.

In the Batch Processing BAE Model, there are two sets of request/response messages. In the first message set, the request message is the criteria for selecting PIV Cardholders, and the response is a list of FASC-Ns that match the criteria. In the second message set, the request is the list of FASC-Ns and desired Backend Attributes, and the response message is Backend Attribute values for each PIV Cardholder returned by the AA.

External BAE Service interface specifications are defined in Sections 4.3 and 5.3.

3.5.1.2 Internal BAE Service

The BAE Internal Service is an intra-agency communications mechanism between an agency system (e.g., RP, AA) and BAE External Service. Agency systems interface only with Internal BAE Services. The Internal BAE service does the following:

- On the BAE Requester side, the BAE Internal Service forwards RP requests to the External BAE Service, and forwards results from the BAE External Service to the RP.
- On the BAE Responder side, the BAE Internal Service receives requests from the External BAE Service, selects the appropriate AA, forwards the request to that AA,

receives results back from the AA, and forwards the results back to the External BAE Service.

The BAE Internal Service interface is out of scope for this document. The participating organization is responsible for implementing the Internal BAE Service and its interface.

3.5.2 Certification Authorities and BAE Metadata Service

To manage trust and connectivity in the BAE network, digital certificates will likely be used to ensure integrity while authenticating BAE Brokers. In addition, every BAE Broker requires certain information about other BAE Brokers with which it will communicate. Certification Authorities and a BAE Metadata Service will likely be part of the BAE architecture. The manner in which they will be implemented depends on the trust and governance model (see Section 3.3), which will be defined by the FICC.

3.5.3 Relying Party (RP)

The RP is the entity that requires Backend Attributes from the applicable authoritative source to satisfy any supported BAE use case. RPs exist within an individual agency infrastructure and are out of scope for this document. Examples of RPs include, but are not limited to the following:

- Physical Access Control System (PACS);
- Logical Access Control System (LACS); and
- Security Guard via a web interface.

3.5.4 Attribute Authority (AA)

For the initial BAE release, the AA is the agency that issued the PIV Card to the PIV Cardholder. The AA is the authoritative source of Backend Attributes for that PIV Cardholder. The applicable AA system responds to Backend Attribute requests by providing the requested information to BAE Brokers as appropriate. Message transactions between AAs and BAE Brokers are internal to each organization and are out of scope for this document. Future BAE releases may support additional authoritative sources.

4 SINGLE PIV CARDHOLDER BAE MODEL

This model will likely be the predominant BAE model used federal government wide. This model uses SAML 2.0 as the underlying protocol to request Backend Attributes for one PIV Cardholder per request – the scope of most BAE use cases. Use of SAML leverages Federal government experience, knowledge (e.g., U.S. E-Authentication Identity Federation), and COTS products.

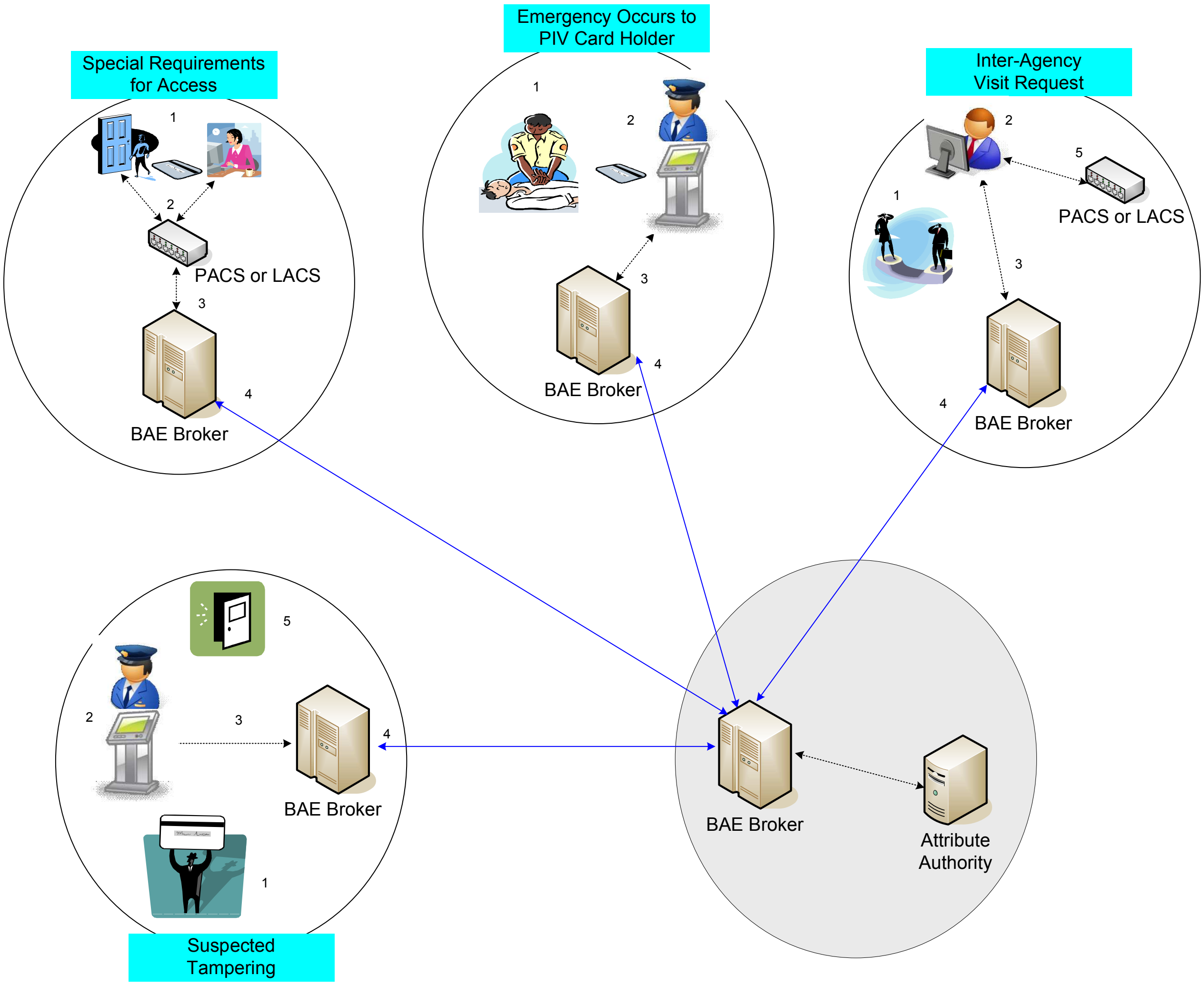
4.1 Overview of Use Cases

Table 4-1 summarizes the initial set of use cases driving the Single PIV Cardholder BAE model. See [BAE Use Cases] for more details. Additional use cases are likely to be developed in the future. Figure 4-1 pictorially summarizes the uses cases.

Table 4-1 BAE Use Case Summary

Use Case #	Use Case Name	Relying Party	PIV Card Present	PIV Cardholder Present	Use Case Summary
1	**Suspected Tampering of the PIV Card**	**Security Guard**	**Yes**	**Yes**	The RP suspects that someone has tampered with physical aspects of the PIV Card (e.g., photograph replaced). The security guard uses BAE to make a final determination by retrieving applicable Backend Attributes, and comparing them to the presented PIV Card, and perhaps the PIV Cardholder.
2	**Special Requirements for Access**	**Security Officer, PACS, LACS**	**Yes**	**Yes**	Subsequent to authentication, the RP needs further information to make authorization or access control decisions. Such information is typically not available from the PIV Card and must be obtained through BAE.
3	**Emergency occurs with PIV Cardholder**	**Security Officer**	**Yes**	**Maybe**	A PIV Cardholder has an emergency (e.g., medical emergency). Those responding use the affected person's PIV Card and BAE to obtain additional information about the person (e.g., contact information) that may help deal with the emergency.
4	**Interagency Visit Request**	**Agency Visitor Management System**	**No**	**No**	An agency hosting a meeting uses BAE to obtain attendee information in advance of a scheduled meeting. The hosting agency then uses retrieved Backend Attributes to provision the meeting site PACS, or any other physical or logical resources applicable to the meeting.

Figure 4-1 BAE Use Cases



4.1.1 Backend Attributes per Use Case

Table 4-2 lists applicable Backend Attributes per use case. However, each Backend Attribute is optional. Additional or fewer Backend Attributes may be used per use case as appropriate (i.e., an RP may request any set of Backend Attributes per use case). See Appendix A for the list of currently supported Backend Attributes.

Table 4-2 Backend Attributes per Use Case

		Use Cases			
	Backend Attribute	Suspected Tampering	Special Requirements for Access	Emergency Occurs to PIV Cardholder	Interagency Visit Request
	PIV 800-73 Data Items				
1	FASC-N	✓	✓	✓	✓
2	FingerprintImage				✓

		Use Cases			
	Backend Attribute	Suspected Tampering	Special Requirements for Access	Emergency Occurs to PIV Cardholder	Interagency Visit Request
3	DigitalSignatureCertificate				✓
4	KeyManagementCertificate				✓
5	CardAuthenticationCertificate				✓
	PIV Card Topology Items				
6	PersonGivenName	✓			✓
7	PersonMiddleName	✓			✓
8	PersonSurName	✓			✓
9	PersonNameSuffixText	✓			✓
10	PersonSexCode	✓			✓
11	PersonOrganizationAssociationCategory	✓	✓		✓
12	OrganizationalAffiliation	✓	✓		✓
13	Photo	✓			✓
14	CardExpirationDate	✓	✓		✓
15	CardIssueDate	✓			✓
16	EmployeeRankText	✓	✓		✓
	Items Not Present on the PIV Card				
17	CHUIDStatus	✓	✓		✓
18	CHUIDStatusDate	✓	✓		✓
19	TelephoneNumber				✓
20	PersonBirthDate				✓
21	PersonCitizenshipFIPS10-4Code				✓
22	USCitizenship				✓
23	PersonSecurityClearanceCode		✓		✓
24	ClearanceDate		✓		✓
25	ClearingAgency		✓		✓
26	CardStatus		✓		✓
27	CardStatusDate		✓		✓
28	DesignatedRole		✓		✓
29	CertificationType		✓		✓
30	CertificationName		✓		✓
31	CertificationDate		✓		✓
	Items Not Present on the PIV Card				
32	CertifyingAuthority		✓		✓
33	EmergencyContactPersonGivenName			✓	✓
34	EmergencyContactPersonSurName			✓	✓
35	EmergencyContactTelephoneNumber			✓	✓
36	EmergencyContactEmail			✓	✓
37	NIPPSectorCode		✓		✓
38	ESFCode		✓		✓

4.2 Detailed Single PIV Cardholder BAE Process Flow

Figure 4-2 details end-to-end Single PIV Cardholder BAE processing. In this depiction, both organizations include AAs, which allows each organization to support individuals visiting from the other organization. Therefore, each BAE Broker may act as both a BAE Responder and a BAE Requester.

Table 4-3 describes each process flow step when an individual from Organization B visits Organization A.

Figure 4-2 End-to-End Single PIV Cardholder BAE Processing

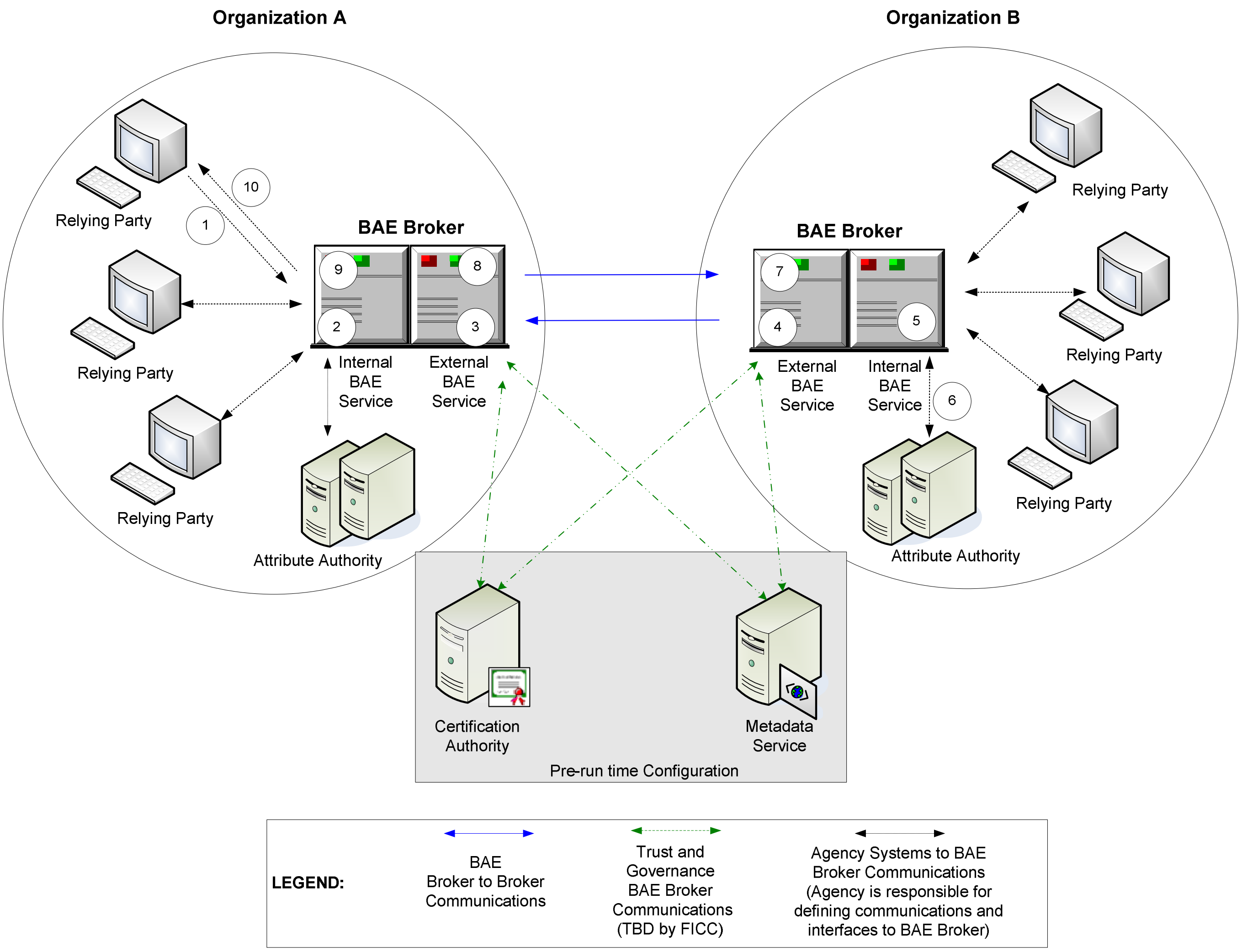


Table 4-3 Detailed Single PIV Cardholder BAE Process Flow

Step	Description
1	Organization A RP obtains FASC-N from the Cardholder Unique Identifier (CHUID) or PIV Authentication Certificate, and sends the FASC-N and the list of requested Backend Attributes to Organization A Internal BAE Service
2	Organization A Internal BAE Service obtains ORG ID (Agency/Sub-agency) from FASC-N. Organization A Internal BAE Service initiates BAE processing by passing FASC-N and ORG ID to Organization A External BAE Service.
3	Organization A External BAE Service: - uses FASC-N ORG ID as key into BAE Metadata to obtain Organization B External BAE Service URL; - creates request to be sent to Organization B External BAE Service; - signs request with Organization A signature private key; - sends Backend Attribute request to Organization B External BAE Service. Request message includes the FASC-N provided by the RP. See Section 4.3.1 for complete details regarding the Single PIV Cardholder BAE request.
4	Upon receiving Backend Attribute request, Organization B External BAE Service: - verifies sender's (Organization A) signature using Organization A signature public key; - sends request to Organization B Internal BAE Service.
5	Upon receiving request, Organization B Internal BAE Service: - selects applicable AA to service the Backend Attribute request; - requests Backend Attributes from selected AA.
6	AA processes the request from Organization B Internal BAE Service and responds with the Backend Attribute information. Organization B Internal BAE Service accepts the Backend Attribute information from AA and sends the information to Organization B External BAE Service.
7	Upon receiving Backend Attribute information, Organization B BAE External Service: - uses Organization A entity ID from the SAML request transaction as key into BAE Metadata to obtain Organization A encryption public key; - creates response to be sent to Organization A External BAE Service; - signs response with Organization B signature private key; - encrypts the assertion portion of the response using Organization A encryption public key; - sends Backend Attribute response to Organization A External BAE Service. Response message includes Backend Attributes provided by the AA. See Section 4.3.2 for complete details regarding the Single PIV Cardholder BAE response.
8	Upon receiving Backend Attribute response, Organization A External BAE Service: - decrypts response using Organization A encryption private key; - verifies sender's (Organization B) signature using Organization B signature public key; - sends response to Organization A Internal BAE Service.
9	Organization A Internal BAE Service returns Backend Attributes to RP.
10	RP stores, uses, or displays returned Backend Attribute values as required.

4.3 Single PIV Cardholder BAE Interface Specification

SAML 2.0 is the underlying protocol of the Single PIV Cardholder BAE Interface Specification. This interface specification consists of a specific subset of the features defined in [SAMLCore] to promote interoperability between Single PIV Cardholder BAE implementations. The Single PIV Cardholder BAE Interface Specification also leverages the SAML SOAP binding defined in [SAML2 Bindings] as a transport mechanism.

SAML 2.0 facilitates the exchange of Backend Attributes through real-time SAML request and response messages directly between endpoints. In BAE, the SAML endpoints are External BAE Services within BAE Brokers. The External BAE Service in the RP domain sends a SAML request. The External BAE Service in the AA domain returns a SAML response. Appendix A defines the current set of Backend Attributes that can be exchanged within BAE. For BAE Single PIV Cardholder Interface Specification purposes:

- A SAML request specifies a set of Backend Attributes for which values are requested; and
- A SAML response provides the requested values, in accordance with applicable security and access control requirements.

The Single PIV Cardholder BAE Interface Specification guides experienced SAML users on how to use SAML 2.0 specifically for BAE purposes. This interface specification does not revise or extend the Organization for the Advancement of Structured Information Standards (OASIS) SAML 2.0 specification. Rather, it simply details how BAE components must use SAML 2.0 for BAE purposes. Where this specification does not explicitly provide SAML guidance, one MUST implement in accordance with SAML 2.0 requirements, as documented by the OASIS standards body.

4.3.1 Request <AttributeQuery> Processing Rules

- `<AttributeQuery>` MUST be communicated using SOAP v 1.1 over HTTP over TLS 1.0.
- `<AttributeQuery>` MUST be signed using `<ds:Signature>.`

Version

- `Version` attribute MUST be set to “2.0”.

Destination

- An `<AttributeQuery>` MUST NOT include the `Destination` attribute.

<Issuer>

- `<Issuer>` MUST NOT be present in an `<AttributeQuery>`.

<ds:Signature>

- `<ds:Signature>` MUST be present in an `<AttributeQuery>`.

<Subject>

- There MUST be exactly one `<Subject>` per `<AttributeQuery>`.
- <NameID> within <Subject> MUST contain a `Format` attribute set to urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified.
- `<NameID>` value MUST be the character representation of FASC-N defined by [FASC-N].

<Attribute>

- `<AttributeQuery>` MUST contain exactly one us:gov:ficc:bae:2008-01:BAESpecVer `<Attribute>` element.
  - us:gov:ficc:bae:2008-01:BAESpecVer `<Attribute>` element MUST be set to “1.0”.
- To request Backend Attributes listed in Appendix A, `<AttributeQuery>` MUST include an `<Attribute>` element for each specific Backend Attribute. The following exception is allowed:
  - As a convenience, to request all Backend Attributes listed in Appendix A, `<AttributeQuery>` MAY exclude Backend Attribute `<Attribute>` elements. That is, an <AttributeQuery> containing only the us:gov:ficc:bae:2008-01:BAESpecVer <Attribute> element and no Backend Attribute `<Attribute>` elements returns all Backend Attributes listed in Appendix A.

- Each <Attribute> element MUST have a Name attribute set to a value defined in Appendix A.
- An <Attribute> element MAY have a NameFormat attribute.
  - If present, NameFormat MUST be set to one of the following values:
    urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified
    urn:oasis:names:tc:SAML:2.0:attrname-format:uri
    urn:oasis:names:tc:SAML:2.0:attrname-format:basic
- Each <Attribute> MAY contain the FriendlyName attribute.
- An <Attribute> MAY contain multiple <AttributeValue>s.
  - If multiple <AttributeValue>s exist, a BAE Responder MUST choose one of the values presented in the request.

4.3.2 Response <Response> Processing Rules

- <Response> MUST be communicated using SOAP v1.1 over HTTP over TLS.

Version

- Version attribute MUST be set to “2.0”.

<Status>

- <StatusMessage> MAY be included in the <Status>.
- <StatusDetail> MAY be included in the <Status>.

<EncryptedAssertion>

- Each <Response> MUST contain no more than one <EncryptedAssertion>.

<Assertion>

- The <Response> element MUST contain exactly one <Assertion>.
- <Assertion> MUST be signed, encrypted, and included in <Response> within <EncryptedAssertion>.

Version

- Version attribute MUST be set to “2.0”.

<Issuer>

- <Issuer> MUST be present and its value MUST be the unique identifier of the BAE Responder issued by the BAE Governing Authority.
- <Issuer> MUST be a Uniform Resource Identifier reference within the BAE Responder domain.

<Subject>

- <Assertion> MUST contain exactly one <Subject> indicating the end user to which <Assertion> pertains.
- <NameID> within <Subject> MUST contain a Format attribute set to urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified.

<Conditions>

- Each <Assertion> MAY contain the optional <Conditions> element.
- If <Conditions> is present in <Assertion>:
  - NotBefore attribute MUST be present.
  - NotAfter attribute MUST be present.

<AttributeStatement>

- <Assertion> MUST contain exactly one <AttributeStatement>.
- <AttributeStatement> MUST contain one or more <Attribute> elements.
- <AttributeStatement> MUST contain exactly one us:gov:ficc:bae:2008-01:BAESpecVer attribute.
  - us:gov:ficc:bae:2008-01:BAESpecVer attribute MUST be set to “1.0”.

- A BAE Responder MUST include an `<Attribute>` element for each requested Backend Attribute.
- Each `<Attribute>` MUST NOT be encrypted.

<Attribute>

- To support the return of multiple values for a Backend Attribute (i.e., a list of values), the `<Attribute>` element MAY contain multiple `<AttributeValue>`s.
- An `<Attribute>` element MAY have a `NameFormat` attribute.
  - If present, `NameFormat` MUST be set to one of the following values:
    urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified
    urn:oasis:names:tc:SAML:2.0:attrname-format:uri
    urn:oasis:names:tc:SAML:2.0:attrname-format:basic
- `Name` attribute in the `<Attribute>` element MUST be set to a value defined in Appendix A.
- `FriendlyName` MAY be used to provide a human readable label for the Backend Attribute.

<AttributeValue>

- For interoperability purposes, xsi:type attribute MUST NOT be used.
- If the attribute value is unknown or otherwise cannot be exchanged <AttributeValue> MUST be blank.

Message Level Status Code

- A `<Response>` message level status code MUST be implemented in accordance with Section 3.2.2.2 in [SAML2 Core] and `<Status>` rules specified above.

4.3.3 Security

Each Single PIV Cardholder BAE message contains a digital signature to protect the integrity of the message and to verify the sender of the message. In addition, each Single PIV Cardholder BAE response message is encrypted to ensure that only the intended recipient can decipher the message and gain access to personally identifiable information. Thus, the Single PIV Cardholder BAE interface specification relies on X.509v3 cryptographic key pairs.

4.3.3.1 BAE Certificates

Each BAE Broker MUST possess a valid BAE certificate to participate in BAE transactions.

4.3.3.2 Digital Signature

The sender MUST sign all Single PIV Cardholder BAE messages, or parts thereof, using its BAE certificate. The signature allows the recipient of the message to authenticate the sender, and confirm that the message has not been altered since the time of signature.

- The recipient MUST authenticate the sender by verifying the signature upon receipt of the message.
- Signature verification MUST use the public key in the sender’s BAE certificate.
- The recipient MUST verify the revocation status of the sender BAE certificate used to sign the message. The recipient SHOULD use one of the following methods for revocation verification:
  - *CDP Extension* – the signature certificate will include a Certificate Revocation List (CRL) Distribution Point (CDP) extension point.
  - *OCSP* – The OCSP URI is available via the `AuthorityInformationAccess` extension.
  - *CRL* – the CRL location (in the directory or web site) can be statically configured into the software, and CRL downloaded periodically.

- If the BAE certificate is revoked or revocation status cannot be determined, the recipient MUST reject the message.

4.3.3.3 Encryption

Encryption ensures that only the intended recipient can decipher the message and gain access to confidential information.

- To protect confidential information, the entire `<Assertion>` in the `<Response>` element MUST be encrypted.
- Encryption MUST use the public key in the intended recipient's BAE certificate.

4.3.4 Metadata

In the Single PIV Cardholder BAE model, BAE Brokers require specific information about each other in order to exchange SAML messages. This section describes the content and format of the metadata required to perform Single PIV Cardholder BAE transactions. This metadata specification conforms to [SAML2 Metadata] and [SAML2 Metadata Ext]. Each BAE metadata file MUST contain metadata for only one BAE participant.

4.3.4.1 <EntityDescriptor>

entityID

- `entityID` MUST be unique.

<**Organization**>

- It is RECOMMENDED that `<Organization>` be present and include either `OrganizationName` or `OrganizationDisplayName`.

<**ContactPerson**>

- It is RECOMMENDED that the `<ContactPerson>` be present and include either `EmailAddress` or `TelephoneNumber` at a minimum.

<**Signature**>

- If the XML root is `<EntityDescriptor>`, then a valid signature enveloped within `<EntityDescriptor>` MUST be included.

4.3.4.2 <AttributeAuthorityDescriptor>

- This element MUST be used by BAE Brokers in the BAE Responder role.

<**KeyDescriptor**>

- `<KeyDescriptor>` MUST be present.
  - `<KeyDescriptor>` MUST indicate the BAE certificate used by the BAE Responder to sign the response message.
- `Use` attribute MUST bet set to "signing".
- `<X509Data>` MUST include the `<X509Certificate>` element populated with the BAE certificate for signing.

<**AttributeService**>

- Exactly one `<AttributeService>` MUST be included in the `<AttributeAuthorityDescriptor>`.

<**NameIDFormat**>

- A `<NameIDFormat>` value of urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified MUST be present in the `<AttributeAuthorityDescriptor>` element.
  - Other name formats MAY be specified.

<AttributeProfile>

- An <AttributeProfile> value of urn:oasis:names:tc:SAML:2.0:attrname-format:basic MUST be present in the <AttributeAuthorityDescriptor> element.

4.3.4.3 <md:RoleDescriptor>

- This element MUST be used by BAE Brokers in the BAE Requester role.

xsi:type

- One xsi:type attribute MUST be present with a value of query:AttributeQueryDescriptorType.
  - The complex type AttributeQueryDescriptorType is derived from the [SAML2 Metadata Ext] abstract RoleDescriptorType complex type.

4.3.4.3.1 <AttributeConsumingService>

isDefault

- If present, the isDefault attribute MUST be set to true.

<KeyDescriptor>

- Two <KeyDescriptor> elements MUST be present.
- One of the <KeyDescriptor> elements MUST indicate the BAE certificate used by the BAE Requester to sign the request.
  - Use attribute MUST indicate "signing".
- The other <KeyDescriptor> element MUST indicate the BAE certificate used by the BAE Requester to encrypt the message.
  - Use attribute MUST indicate "encryption".
- <X509Data> within each <KeyDescriptor> MUST include the <X509Certificate> element populated with the corresponding BAE certificate.

<NameIDFormat>

- A <NameIDFormat> value of urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified MUST be present in the <AttributeAuthorityDescriptor> element.
  - Other name formats MAY be specified.

<ServiceDescription>

- <ServiceDescription> MAY be included in the <AttributeConsumingService> element.

<RequestedAttribute>

- BAE Requester MUST include a <RequestedAttribute>
- Name Attribute in the <RequestedAttribute> element MUST be set to "us:gov:ficc:bae:2008-01:BAESpecVer".

4.3.5 XML Samples

4.3.5.1 Sample Request

Figure 4-3 Sample AttributeQuery

```
<samlp:AttributeQuery xmlns:samlp:=”…” xmlns:saml=”…” xmlns:ds=”…”
ID=”_6c3a4f8b9c2d” Version=”2.0” IssueInstant=”2008-03-27T08:41:00Z”>
            <ds:Signature> … </ds:Signature>
            <saml:Subject xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
                        <saml:NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-
                                                format:unspecified">a9c16e8616880860
                        </saml:NameID>
            </saml:Subject>
            <saml:Attribute Name="us:gov:ficc:bae:2008-01:BAESpecVer”
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
                        <saml:AttributeValue>1.0</saml:AttributeValue>
            </saml:Attribute>
            <saml:Attribute Name=" nc:PersonSexCode”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

            </saml:Attribute>
            <saml:Attribute Name="us:gov:ficc:bae:2008-01:CardExpirationDate”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

            </saml:Attribute>
            <saml:Attribute Name="us:gov:ficc:bae:2008-01: CardStatus”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

            </saml:Attribute>
            <saml:Attribute Name="us:gov:ficc:bae:2008-01:USCitizenship”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

            </saml:Attribute>
            <saml:Attribute Name="us:gov:ficc:bae:2008-01:ClearingAgency”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">

            </saml:Attribute>
</samlp:AttributeQuery>
```

4.3.5.2 Sample Response[3]

Figure 4-4 Sample Response

```
<samlp:Response xmlns:samlp=”…” xmlns:saml=”…” xmlns:ds=”…”
ID=”_6c3a4f8b9c2d” Version=”2.0” IssueInstant=”2008-03-27T08:42:00Z”>
        <ds:Signature> … </ds:Signature>
        <Status>
                 <StatusCode Value=”urn:oasis:names:tc:SAML:2.0:status:Success”/>
        </Status>
        <saml:Assertion>
        <saml:Issuer>https://www.example-agency.gov/BAE</saml:Issuer>
        <ds:Signature>…</ds :Signature>
        <saml:Subject>
                <saml:NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-
                                format:unspecified">a9c16e8616880860
                </saml:NameID>
        </saml:Subject>
        <saml:Conditions xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
                NotBefore=”2008-03-27T08:41:00Z” NotAfter=”2008-03-
        27T08:43:00Z”>
        </saml:Conditions>
        <saml:AttributeStatement>
                <saml:Attribute Name="us:gov:ficc:bae:2008-01:BAESpecVer”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                format:basic">
                        <saml:AttributeValue>1.0</saml:AttributeValue>
                </saml:Attribute>
                <saml:Attribute Name=" nc:PersonSexCode”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                format:basic">
                        <saml:AttributeValue>M</saml:AttributeValue>
                </saml:Attribute>
                <saml:Attribute Name="us:gov:ficc:bae:2008-01:CardExpirationDate”
                NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
                        <saml:AttributeValue>2009-11-25</saml:AttributeValue>
                </saml:Attribute>
                <saml:Attribute Name="us:gov:ficc:bae:2008-01: CardStatus”
                        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                format:basic">
                        <saml:AttributeValue>PRO</saml:AttributeValue>
                </saml:Attribute>


(Continued)
```

[3] An actual attribute response contains an <EncryptionAssertion>, which has been omitted here for readability purposes.

```
<saml:Attribute Name="us:gov:ficc:bae:2008-01:USCitizenship”
          NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
<saml:AttributeValue>1</saml:AttributeValue>
</saml:Attribute>
<saml:Attribute Name=" us:gov:ficc:bae:2008-01:ClearingAgency”
          NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
<saml:AttributeValue>0901</saml:AttributeValue>
<saml:AttributeValue>0100</saml:AttributeValue>
<saml:AttributeValue>1341</saml:AttributeValue>
<saml:AttributeValue>19BE</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
</saml:Assertion>
</samlp:Response>
```

5 BATCH PROCESSING BAE MODEL

The Single PIV Cardholder BAE model supports the use cases defined in Section 4.1 by using SAML for real-time retrieval of Backend Attributes for one PIV Cardholder at a time. SAML is designed for such a purpose (i.e., one subject per request/response pair). However, agencies may require Backend Attributes for multiple PIV Cardholders in a single session (Batch Processing BAE). The current SAML specification does not support batch processing. SPML 2.0 does support batch processing, and therefore is the underlying protocol of the Batch Processing BAE model.

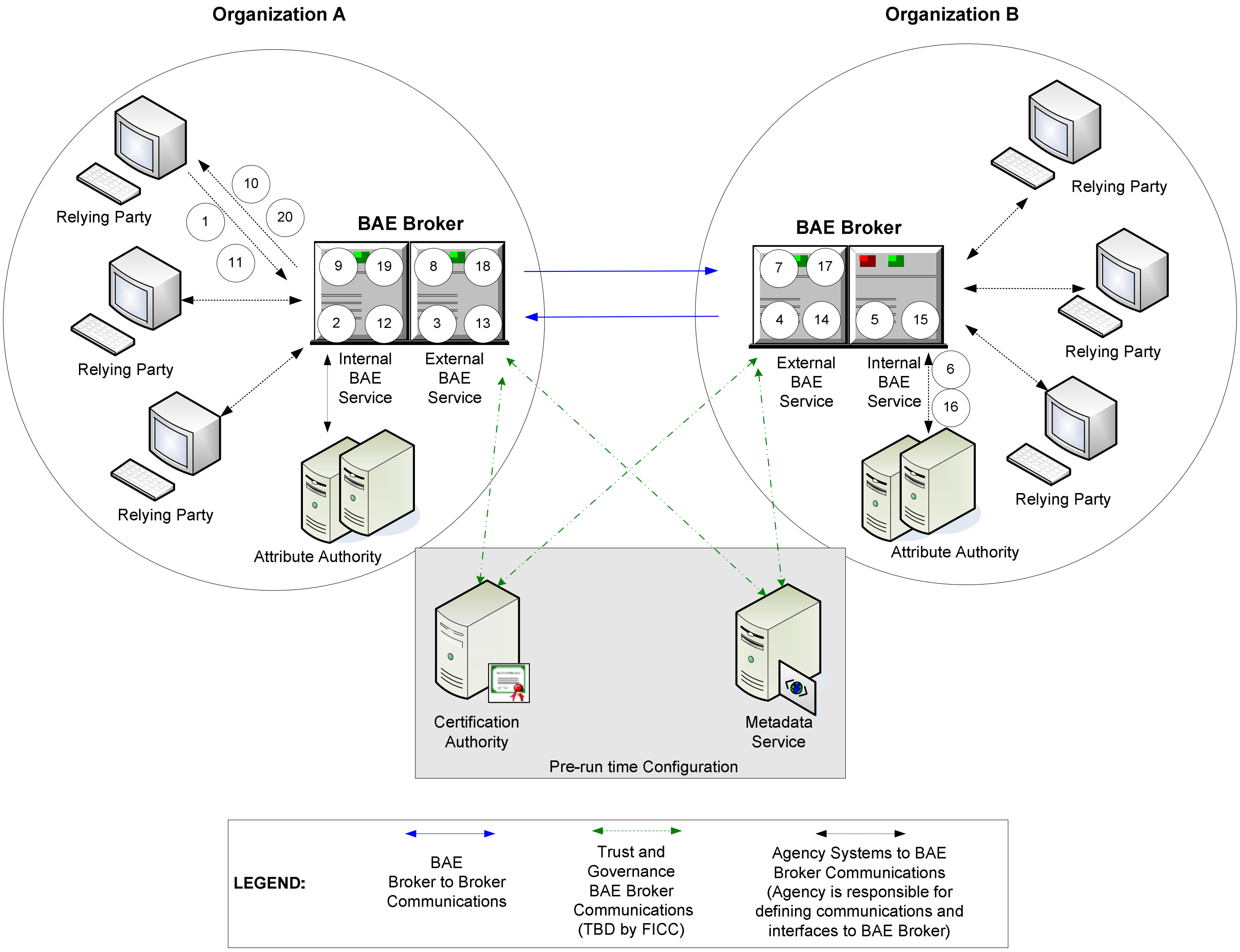
5.1 Example Batch Processing BAE Use Case

Batch Processing BAE is useful in many different use cases. One particularly important use case is Federal Emergency Management Agency (FEMA) first responders who use handheld devices to authenticate PIV Cardholders at incident sites. In this use case, FEMA's handheld devices are carried into field environments where communication facilities may either be limited or unavailable. As a result, FEMA may use a Batch Processing BAE server to load handheld devices with Backend Attributes of multiple PIV Cardholders prior to field deployment. In addition, FEMA may use a Batch Processing BAE server to quickly update Backend Attribute information in the handheld devices after initial loading. To update handheld devices with updated Backend Attributes, FEMA requests Backend Attributes of PIV Cardholders whose records have been modified since a specified date.

5.2 Detailed Batch Processing BAE Flow

Two SPML 2.0 request/response message exchanges are required to complete Batch Processing BAE. The exchanges are described in Section 5.3.1, *Batch Updated PIV Cardholder Search Message Exchange,* and Section 5.3.2, *Batch Backend Attribute Retrieval Message Exchanges*. Batch Processing BAE message exchanges share similar process flows. Figure 5-1 provides a detailed illustration of both message exchanges. Table 5-1 (Steps 1-10) and Table 5-2 (Steps 11-20) detail each message exchange.

Figure 5-1 End-to-End Batch Processing BAE Flow



Batch Updated PIV Cardholder Search returns all PIV Cardholders whose records have been modified since the RP-specified date, and whose records share common attributes or attribute values. See Section 5.3.1 for the interface specification.

Table 5-1 Batch Updated PIV Cardholder Process Flow

Step	Description
1	Organization A RP requests a batch search from Organization A Internal BAE Service. RP provides the organization (e.g., Organization B) and search parameters (i.e., distinguishing attribute values (e.g., ESF=3), and an 'updated since' date).
2	Organization A Internal BAE Service obtains ORG ID (Agency/Sub-agency) from RP request. Organization A Internal BAE Service initiates a Batch Updated PIV Cardholder Search by passing ORG ID and search parameters to Organization A External BAE service.
3	Organization A External BAE Service: - uses ORG ID as key into BAE Metadata to obtain Organization B External BAE Service URL; - creates request to be sent to Organization B External BAE Service; - signs request with Organization A signature private key; - sends BAE Updates Search request to Organization B External BAE Service. Request message includes the search parameters provided by the RP. See Section 5.3.1.1 for complete details regarding the Batch Updated PIV Cardholder Search request.
4	Upon receiving BAE Updates Search request, Organization B External BAE Service: - verifies sender's (Organization A) signature using Organization A signature public key; - sends request to Organization B Internal BAE Service.
5	Upon receiving request, Organization B Internal BAE Service: - selects applicable AA(s) to service the BAE Updates Search request (PIV Cardholders listed in the request may require different AAs); - requests a list of FASC-Ns for PIV Cardholders meeting the search parameters from the AA(s).
6	AA(s) processes the request from Organization B Internal BAE Service and responds with the list of FASC-Ns for PIV Cardholders meeting the search parameters. Organization B Internal BAE Service accepts the Backend Attribute information from AA and sends the information to Organization B External BAE Service.
7	Upon receiving Backend Attribute information, Organization B BAE External Service: - uses Organization A signing certificate from the SPML request transaction as key into BAE Metadata to obtain Organization A encryption public key; - creates response to be sent to Organization A External BAE Service; - signs response with Organization B signature private key; - encrypts the response using Organization A encryption public key; - sends BAE Updates Search response to Organization A External BAE Service See Section 5.3.1.2 for complete details regarding the Batch Updated PIV Cardholder Search response.
8	Upon receiving BAE Updates Search response, Organization A External BAE Service: - decrypts response using Organization A encryption private key; - verifies sender's (Organization B) signature using Organization B signature public key; - sends response to Organization A Internal BAE Service
9	Organization A Internal BAE Service returns list of FASC-Ns to RP
10	RP examines the list of FASC-Ns returned from the search and trims (i.e., selects all, none, or a sub set of returned FASC-Ns) as necessary. RP optionally invokes a Batch Backend Attribute Retrieval request to retrieve Backend Attributes.

Batch Backend Attribute Retrieval message exchange obtains attributes for a list of PIV Cardholders. The list of PIV Cardholders may be obtained using the *Batch Updated PIV Cardholder Search*, or it may be obtained through other means. See Section 5.3.2 for the interface specification.

Table 5-2 Batch Backend Attribute Retrieval Process Flow

Step	Description
11	Organization A RP requests a batch Backend Attribute retrieval from Organization A Internal BAE Service. RP provides the organization from which to retrieve the attributes, a list of PIV Cardholder FASC-Ns, and requested attributes.
12	Organization A Internal BAE Service obtains ORG ID (Agency/Sub-agency) from RP request. Organization A Internal BAE Service initiates Batch Backend Attribute Retrieval by passing ORG ID, FASC-Ns, and requested attributes to Organization A External BAE service.
13	Organization A External BAE Service: - uses ORG ID as key into BAE Metadata to obtain Organization B External BAE Service URL; - creates Batch Backend Attribute Retrieval request to be sent to Organization B External BAE Service; - signs request with Organization A signature private key; - sends Batch Backend Attribute Retrieval request to Organization B External BAE Service. Request message includes the search parameters provided by the RP. See Section 5.3.2.1 for complete details regarding the Batch Backend Attribute Retrieval request.
14	Upon receiving Batch Backend Attribute Retrieval request, Organization B External BAE Service: - verifies sender's (Organization A) signature using Organization A signature public key; - sends request to Organization B Internal BAE Service.
15	Upon receiving request, Organization B Internal BAE Service: - selects applicable AA(s) to service the Batch Backend Attribute Retrieval request; - requests the Backend Attribute information for the PIV Cardholders listed in the Batch Backend Attribute Retrieval request.
16	AA(s) processes the request from Organization B Internal BAE Service and responds with the Backend Attribute information. Organization B Internal BAE Service accepts the Backend Attribute information from AA and sends the information to Organization B External BAE Service.
17	Upon receiving Backend Attribute information, Organization B BAE External Service: - uses Organization A signing certificate from the SPML request transaction as key into BAE Metadata to obtain Organization A encryption public key; - creates response to be sent to Organization A External BAE Service; - signs response with Organization B signature private key; - encrypts the response using Organization A encryption public key; - sends Batch Backend Attribute Retrieval response to Organization A External BAE Service Response message includes Backend Attribute information provided by the AA(s). See Section 5.3.2.2 for complete details regarding Batch Backend Attribute Retrieval response.
18	Upon receiving Batch Backend Attribute Retrieval response, Organization A External BAE Service: - decrypts response using Organization A encryption private key; - verifies sender's (Organization B) signature using Organization B signature public key; - sends response to Organization A Internal BAE Service
19	Organization A Internal BAE Service returns Backend Attribute information to RP
20	RP stores, uses, or displays returned Backend Attribute values as required

5.3 Batch Processing BAE Interface Specification

This section specifies how to implement Batch Processing BAE using SPML 2.0. The Batch Processing BAE Interface Specification does not revise or extend [SPML2]. Rather, it simply details how BAE components must use SPML 2.0 for Batch Processing BAE purposes. Where this specification does not explicitly provide SPML 2.0 guidance, one must implement in accordance with SPML 2.0 requirements as documented by the OASIS standards body. See Appendix A for the list of currently supported Backend Attributes.

5.3.1 Batch Updated PIV Cardholder Search Message Exchange

Batch Updated PIV Cardholder Search message exchange uses SPML-based <updatesRequest> and <updatesResponse> messages. The interface specification for this message set follows.

5.3.1.1 <updatesRequest>

The BAE Requester uses the `<updatesRequest>` operation to request a list of PIV Cardholders whose records have been recently updated.

- The group of desired PIV Cardholders can be identified using the [XPATH] query located within the `<select>` element.
- The `path` attribute within the `<select>` element SHOULD include at least one of the following attributes from Appendix A:
    - `us:gov:ficc:bae:2008-01:ESFCode`
    - `us:gov:ficc:bae:2008-01:NIPPSectorCode`
  - For example, `path='Person/us:gov:ficc:bae:2008-01:ESFCode="3"'` directs the BAE Responder to search within subjects that have an Emergency Support Function attribute with a value of "3".
- `<updatesRequest>` MUST be wrapped in SOAP version 1.1 [SOAP].
- The `<updatesRequest>` message MUST be signed using the BAE Requester's private signing key.
  - The signature MUST be contained in the SOAP Header in accordance with WS-Security version 1.1 [WS-Security].
- Refer to [NRP] and [NIPP] for the codes to use when making this request.

Figure 5-2 provides an example of the `<updatesRequest>` message.

Figure 5-2 Sample Batch Processing BAE updatesRequest

```
<updatesRequest requestID=”145” updatedSince=”20040501115900">
        <query scope=”subTree” targetID=”target2” >
        <select path=’Person/us:gov:ficc:bae:2008-01:ESFCode=”3”’
        namespaceURI=”http://www.w3.org/TR/xpath20” />
        </query>
</updatesRequest>
```

5.3.1.2 <updatesResponse>

The BAE Responder uses the `<updatesResponse>` operation in response to an `<updatesRequest>`. `<updatesResponse>` returns a list of PIV Cardholder who are members of a specified community and whose FASC-N records have been updated since a requested date.

- FASC-N MUST be formatted in a character representation defined by [FASC-N].
- Every `<psoID>` element MUST contain an `ID` attribute whose value is the FASC-N of one of the PIV Cardholders meeting the search criteria from the corresponding `<updatesRequest>.`
- `<updatesResponse>` MUST be wrapped in SOAP version 1.1 [SOAP].
- The `<updatesResponse>` message MUST be signed using the BAE Responder's private signing key.
  - The signature MUST be contained in the SOAP Header in accordance with [WS-Security].
- The `<updatesResponse>` message MUST be encrypted using the BAE Requester's public encryption key in accordance with [WS-Security].

Figure 5-3 provides an example of the `<updatesResponse> message`.

Figure 5-3 Sample Batch Processing BAE updatesResponse

```
<updatesResponse requestID=”145” status=“success">
        <update timestamp="20050704115900" updateKind="modify">
        <psoID ID=”FASC-N 1” targetID=“target2”/>
        </update>
        <update timestamp="20050704115900" updateKind="modify">
        <psoID ID=” FASC-N 2” targetID=“target2”/>
        </update>
        <update timestamp="20050704115900" updateKind="modify">
        <psoID ID=” FASC-N 3” targetID=“target2”/>
        </update>
        <update timestamp="20050704115900" updateKind="modify">
        <psoID ID=” FASC-N 4” targetID=“target2”/>
        </update>
</updatesResponse>
```

5.3.2 Batch Backend Attribute Retrieval Message Exchange

Batch Backend Attribute Retrieval message exchange uses SPML-based <lookupRequest> and <lookupResponse> messages. The interface specification for this message set follows.

5.3.2.1 <lookupRequest>

The BAE Requester uses the `<lookupRequest>` operation to request Backend Attributes for a list of PIV Cardholders. The list of PIV Cardholders is usually obtained from an `<updatesResponse>` but can be obtained by other means.

- FASC-N MUST be formatted in a character representation as defined by [FASC-N].
- Each `ID` attribute of the `<psoID>` element MUST contain a FASC-N of a PIV Cardholder.
- The `<lookupRequest>` message MUST be signed using the BAE Requester's private signing key.
  - The signature MUST be contained in the SOAP Header in accordance with [WS-Security].

Figure 5-4 provides an example of the `<lookupRequest>` message.

Figure 5-4 Sample Batch processing BAE lookupRequest

```
<batchRequest processing=”sequential” onError=”exit”>
        <spml:lookupRequest returnData = "spml:everything"
        xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
                <spml:psoID ID=" FASC-N 1" targetID="target2"/>
        </spml:lookupRequest>
        <spml:lookupRequest returnData = "spml:everything"
        xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
                <spml:psoID ID=" FASC-N 2" targetID="target2"/>
        </spml:lookupRequest>
        <spml:lookupRequest returnData = "spml:everything"
        xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
                <spml:psoID ID=" FASC-N 4" targetID="target2"/>
        </spml:lookupRequest>
</batchRequest>
```

5.3.2.2 <lookupResponse>

The BAE Responder uses the `<lookupResponse>` operation in response to a `<lookupRequest>`. `<lookupResponse>` is used to return Backend Attributes for a batch of PIV Cardholders.

- FASC-N MUST be formatted in a character representation as defined by [FASC-N].
- The `ID` attribute of the `<psoID>` element MUST be the FASC-N of the PIV Cardholder whose data is contained within the `<psoID>.`
- The `<data>` element within the `<psoID>` element SHOULD contain a `<bae:FirstResponder>` object.
  - If other objects are required or developed, they SHOULD be shared with the FICC.
- The `<lookupResponse>` message MUST be signed using the BAE Responder’s private signing key. The signature MUST be contained in the SOAP Header in accordance with [WS-Security].
- The `<lookupResponse>` message MUST be encrypted using the BAE Requester’s public encryption key in accordance with [WS-Security].

Section 5.3.3 provides an XML schema that defines a `<bae:FirstResponder>` object (or complex element) to relay user attribute information for use within BAE SPML. Figure 5-5 provides an example of the `<lookupResponse>` message.

Figure 5-5 Sample Batch Processing BAE lookupResponse

```
<batchResponse status=“success">
        <spml:lookupResponse status="spml:success"
        xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
                <spml:psoID ID=" FASC-N 1" targetID="target2"/>
                        <spml:data>
                                <bae:FirstResponder>
                                        <saml:Attribute Name="us:gov:ficc:bae:2008-
01:FASC-N”
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                format:basic">

        <saml:AttributeValue>XX</saml:AttributeValue>
                                        </saml:Attribute>
                                        <saml:Attribute Name="us:gov:ficc:bae:2008-
01:ESFCode”
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                format:basic">
                                        <saml:AttributeValue>3</saml:AttributeValue>
                                        <saml:AttributeValue>12</saml:AttributeValue>
                                        </saml:Attribute>
                                </bae:FirstResponder>
                </spml:data>
        </spml:lookupResponse>
        <spml:lookupResponse status="spml:success"
        xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
                <spml:psoID ID=" FASC-N 2" targetID="target2"/>
                        <spml:data>
                                <bae:FirstResponder>


(continued)
```

```
<saml:Attribute Name="us:gov:ficc:bae:2008-01:FASC-N”
                NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
                    format:basic">

    <saml:AttributeValue>XX</saml:AttributeValue>
                    </saml:Attribute>
                    <saml:Attribute Name="us:gov:ficc:bae:2008-
01:ESFCode”
    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
        format:basic">
                    <saml:AttributeValue>3</saml:AttributeValue>
                    <saml:AttributeValue>12</saml:AttributeValue>
                    </saml:Attribute>
                </bae:FirstResponder>
        </spml:data>
    </spml:lookupResponse>


    <spml:lookupResponse status="spml:success"
    xmlns:spml="urn:oasis:names:tc:SPML:2:0" >
        <spml:psoID ID=" FASC-N 4" targetID="target2"/>
            <spml:data>
                <bae:FirstResponder>
                    <saml:Attribute Name="us:gov:ficc:bae:2008-
01:FASC-N”
    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
        format:basic">

    <saml:AttributeValue>XX</saml:AttributeValue>
                    </saml:Attribute>
<saml:Attribute Name="us:gov:ficc:bae:2008-01:ESFCode”
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
            format:basic">
                    <saml:AttributeValue>3</saml:AttributeValue>
                    <saml:AttributeValue>12</saml:AttributeValue>
                    </saml:Attribute>
                </bae:FirstResponder>
        </spml:data>
    </spml:lookupResponse>
</batchResponse>
```

5.3.3 First Responder Schema

Figure 5-6 defines the XML object (complex element) for `<bae:FirstResponder>`. Processing rules for the first responder `<bae:FirstResponder>` complex element are as follows:

<Attribute>

- To support the return of multiple values for a Backend Attribute (i.e., a list of values), the `<Attribute>` element MAY contain multiple `<AttributeValue>`s.
- An `<Attribute>` element MAY have a `NameFormat` attribute.
  - If present, `NameFormat` MUST be set to one of the following values:
    urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified
    urn:oasis:names:tc:SAML:2.0:attrname-format:uri
    urn:oasis:names:tc:SAML:2.0:attrname-format:basic
- The `Name` attribute of each `<saml:Attribute>` element SHOULD be one of the following attributes as defined is Appendix A:
  - us:gov:ficc:bae:2008-01:FASC-N
  - us:gov:ficc:bae:2008-01:ESFCode
  - us:gov:ficc:bae:2008-01:NIPPSectorCode
- `FriendlyName` MAY be used to provide a human readable label for the Backend Attribute.

<AttributeValue>

- For interoperability purposes, xs:Type MUST not be used.
- If the attribute value is unknown or otherwise cannot be exchanged `<AttributeValue>` MUST be empty.

Figure 5-6 Sample Batch Processing BAE First Responder XML

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="us:gov:ficc:bae:2008-01"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:bae=" us:gov:ficc:bae:2008-01">
        <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
        schemaLocation="http://docs.oasis-open.org/security/saml/v2.0/saml-schema-
assertion-2.0.xsd"/>
        <element name="FirstResponder" type="us:gov:ficc:bae:2008-
        01:FirstResponderType"/>
        <complexType name="FirstResponderType">
                <complexContent>
                        <sequence>
                                <element ref="saml:Attribute" minOccurs=”0”
maxOccurs=”unbounded”/>
                        </sequence>
                </complexContent>
        </complexType>
</schema>
```

5.4 Metadata

In the Batch Processing BAE model, BAE Brokers require specific information about each other in order to exchange SPML messages. Currently, [SPML2] does not include a metadata specification. Therefore, Batch Processing BAE will initially require manual updating of metadata on a per connection basis. When SPML standards provide a metadata specification, the Batch Processing BAE Interface Specification will likely be revised to use it. This section describes the likely content of metadata required to perform Batch Processing BAE transactions.

- BAE Requestor:
  - BAE Responder URI endpoint
  - BAE Responder Signature Certificate
  - BAE Responder <spml:data> extensible element schema URI within the <spml:lookupResponse>
  - Organization (Optional)
  - Technical Contact Person (Optional)

- BAE Responder:
  - Batch Requestor Signature Certificate
  - Batch Requestor Encryption Certificate
  - Organization (Optional)
  - Technical Contact Person (Optional)

5.5 Security

Each Batch Processing BAE message contains a digital signature to protect the integrity of the message and to verify the sender of the message. In addition, each Batch Processing BAE response message is encrypted to ensure that only the intended recipient can decipher the message and gain access to personally identifiable information. Thus, the Batch Processing BAE interface specification relies on X.509v3 cryptographic key pairs.

5.5.1 BAE Certificates

Each BAE Broker MUST possess a valid BAE certificate to participate in BAE transactions.

5.5.2 Digital Signature

The sender MUST sign all BAE messages, or parts thereof, using its BAE certificate. The signature allows the recipient of the message to authenticate the sender, and confirm that the message has not been altered since the time of signature.

- The recipient MUST authenticate the sender by verifying the signature upon receipt of the message.
- Signature verification MUST use the public key in the sender's BAE certificate.
- The recipient MUST verify the revocation status of the sender BAE certificate used to sign the message. The recipient SHOULD use one of the following methods for revocation verification:
  - *CDP Extension* – the signature certificate will include a Certificate Revocation List (CRL) Distribution Point extension point.
  - *OCSP* – The OCSP URI is available via the `AuthorityInformationAccess` extension.
  - *CRL* – the CRL location (in the directory or web site) can be statically configured into the software, and CRL downloaded periodically.

- If the BAE certificate is revoked or revocation status cannot be determined, the recipient MUST reject the message.

5.5.3 Encryption

Encryption ensures that only the intended recipient can decipher the message and gain access to confidential information.

- To protect confidential information, the entire `<lookupResponse>` message MUST be encrypted and the entire `<updatesResponse>` message MUST be encrypted.
- Encryption MUST use the public key in the intended recipient's BAE certificate.

6 Appendix A: Supported Backend Attributes

Table A-1 lists the current set of Backend Attributes that the RP may request. Future versions of this document may support additional Backend Attributes, as government-wide experience and feedback dictates.

No Backend Attribute is mandatory. The RP can request any combination of Backend Attributes listed in Table A-1 – regardless of the use case. However, certain Backend Attributes are ostensibly applicable to certain use cases and probably should be used in those cases.

Each Backend Attribute name is a unique URI. The beginning portion of the URI is us:gov:ficc:bae. The middle portion of the URI is the date of inclusion into this interface specification. The last portion of the URI is a "human-friendly" Backend Attribute name. National Information Exchange Model (NIEM) core element names are used where practical (i.e. "prefix:nc" and "namespace: http://niem.gov/niem/niem-core/2.0"). Backend Attribute data types are simple data types (i.e., no sub elements). List based Backend Attributes are multi-value format.

NIEM provides a number of Extensible Markup Language (XML) namespaces and namespace prefixes for XML schemas (or data models). Many NIEM data elements are complex data types comprising multiple sub-elements that are many layers deep. The typical agency will not require NIEM data elements and will find the simple structure of Backend Attributes listed in Table A-1 easier to work with.

Table A-1 Supported Backend Attributes

#	Backend Attribute Name	Data Type	Format	Notes
1	us:gov:ficc:bae:2008-01:FASC-N	base64Binary		Per [FASC-N]
2	us:gov:ficc:bae:2008-01:FingerprintImage	base64Binary		Base64 encoded fingerprint image in the JPEG 2000 format
3	us:gov:ficc:bae:2008-01:DigitalSignatureCertificate	base64Binary		Base64 encoded X.509 v3 certificate
4	us:gov:ficc:bae:2008-01:KeyManagementCertificate	base64Binary		Base64 encoded X.509 v3 certificate
5	us:gov:ficc:bae:2008-01:CardAuthenticationCertificate	base64Binary		Base64 encoded X.509 v3 certificate
6	nc:PersonGivenName	string	<= 60 char	
7	nc:PersonMiddleName	string	<= 60 char	
8	nc:PersonSurName	string	<= 60 char	
9	nc:PersonNameSuffixText	string	<= 12 char	
10	nc:PersonSexCode	string	1 char	M or F
11	us:gov:ficc:bae:2008-01:PersonOrganizationAssociationCategory	integer	1 char	POA Code Per [FASC-N]
12	us:gov:ficc:bae:2008-01:OrganizationalAffiliation	integer	4 char	Agency Code Per [NIST 800-87]
13	us:gov:ficc:bae:2008-01:Photo	base64Binary		JPEG2000
14	us:gov:ficc:bae:2008-01:CardExpirationDate	date	YYYY-MM-DD	e.g., 2008-04-07

#	Backend Attribute Name	Data Type	Format	Notes
15	us:gov:ficc:bae:2008-01:CardIssueDate	date	YYYY-MM-DD	e.g., 2008-04-07
16	nc:EmployeeRankText	string	3 char	
17	us:gov:ficc:bae:2008-01:CHUIDStatus	string	3 char	Allowed values[4]: ACT - Active, SUS - Suspended TER - Terminated
18	us:gov:ficc:bae:2008-01:CHUIDStatusDate	date	YYYY-MM-DD	e.g., 2008-04-07
19	nc:TelephoneNumber	string	###-###-####	e.g., 202-555-1234
20	nc:PersonBirthDate	date	YYYY-MM-DD	e.g., 2008-04-07
21	nc:PersonCitizenshipFIPS10-4Code	string	2 char	Per [FIPS 10-4]
22	us:gov:ficc:bae:2008-01:USCitizenship	boolean	1 char	{true, false, 1, 0}
23	nc:PersonSecurityClearanceCode	string	<= 60 char	May be a list
24	us:gov:ficc:bae:2008-01:ClearanceDate	date	YYYY-MM-DD	May be a list
25	us:gov:ficc:bae:2008-01:ClearingAgency	string	4 char	Agency Code Per [NIST 800-87] May be a list
26	us:gov:ficc:bae:2008-01:CardStatus	string	3 char	PRO-Provisional or PER-Permanent
27	us:gov:ficc:bae:2008-01:CardStatusDate	date	YYYY-MM-DD	e.g., 2008-04-07
28	us:gov:ficc:bae:2008-01:DesignatedRole	string	<=60 char	May be a list
29	us:gov:ficc:bae:2008-01:CertificationType	string	<= 60 char	May be a list
30	us:gov:ficc:bae:2008-01:CertificationName	string	<= 60 char	May be a list
31	us:gov:ficc:bae:2008-01:CertificationDate	date	YYYY-MM-DD	e.g., 2008-04-07 May be a list
32	us:gov:ficc:bae:2008-01:CertifyingAuthority	string	<= 60 char	May be a list
33	us:gov:ficc:bae:2008-01: EmergencyContactPersonGivenName	string	<=60 char	May be a list
34	us:gov:ficc:bae:2008-01: EmergencyContactPersonSurName	string	<=60 char	May be a list
35	us:gov:ficc:bae:2008-01:EmergencyContactTelephoneNumber	string	###-###-####	e.g., 202-555-1234 May be a list
36	us:gov:ficc:bae:2008-01:EmergencyContactEmail	string	<=70 char	May be a list
37	us:gov:ficc:bae:2008-01:NIPPSectorCode	integer	<= 2 char	Per [NIPP] May be a list
38	us:gov:ficc:bae:2008-01:ESFCode	integer	<=2 char	Per [NRP] May be a list

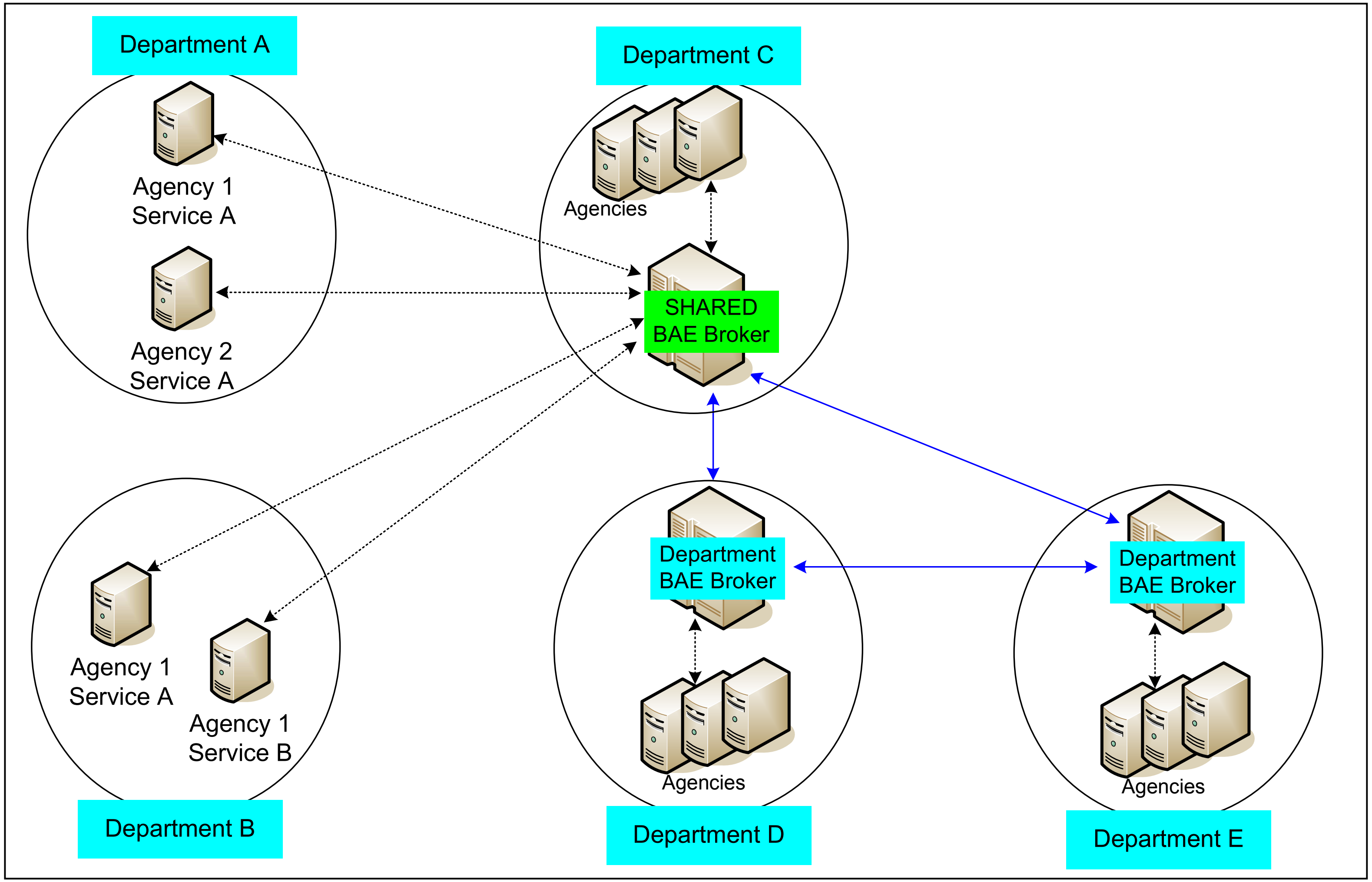
[4] This field must be consistent with [GSA USAccess] allowable ChuidStatus data element values.

7 Appendix B: Shared BAE Broker

A BAE Broker provider may share their BAE Broker with other organizations. Shared BAE Brokers allow other organizations, particularly small agencies, to leverage existing infrastructure, likely resulting in benefits such as reduced BAE costs (e.g., implementation, operational). A government or commercial entity may provide Shared BAE Brokers. Figure B-1 depicts a Shared BAE Broker co-existing with organization-specific BAE Brokers[5]. The introduction and operation of Shared BAE Brokers into the BAE architecture do not require any special BAE architecture or BAE interface specification considerations.

[5] In this example, the organization-specific level is the Department level. See Section 3.4 for more details.

Figure B-1 BAE Shared Architecture Example



Appendix C: BAE Use Case Origin

> Please note that this Appendix is for historical information purposes only. The content was published early in the BAE concept definition process (8/27/2007) as *Use Cases for Defining Back-End Attribute Exchange*. Its content has been overtaken by subsequent HSPD-12 AWG efforts – see Sections 4.1 and 5.1 for more complete and up-to-date BAE use case discussions.

BAE use cases are compiled from several different agency representatives providing specific scenarios. Each use case was generalized to better encompass the wide spectrum of specific use cases. The use cases presented illustrate a practical example of agencies requiring further information which can be retrieved through back-end systems. Consensus on the use cases and the required attributes they define will in turn lead to development of structured interfaces for implementing BAE. Any additional use cases will undergo an acceptance process and be added to future revisions of this document.

Case: Suspected Tampering of the PIV Credential

The PIV credential supports multiple assurance levels and corresponding authentication methods as defined in FIPS 201. If at any time during an authentication transaction a human guard suspects the PIV credential has been tampered with, the human guard may opt to use back-end attribute exchange to support their decision. The relying party, the human guard in this instance, may choose to retrieve PIV card attributes from the PIV card issuing agency. These attributes can be used to provide another level of confidence in the identity of the PIV cardholder.

Variants

- None Determined

Sample Attributes

- Photo
- Printed Information
- Name
- Agency
- Expiration Date

Case: Special Requirements for Access

After the PIV authentication process occurs a relying party may need further cardholder information to make access based decisions. In some facilities, access to nuclear materials may require a PIV cardholder to have previously taken appropriate training and/or certification. These data elements necessary to make access level decisions will generally not be found on the card and must be acquired through back-end transactions. Upon return of the requested attributes, the relying party may make an informed authorization decision.

Variants

- Subset Case #1: A location requires a specific clearance to enter
- Subset Case #2: A location requires special training or certifications to enter
- Subset Case #3: A location requires special emergency responder credential requirements to enter
- Subset Case #4: A location requires a cardholder to be a member of a special role to gain access

- Subset Case #5: A location requires a specific NACI indicator value to gain entrance

Sample Attributes

- Clearance Data
- Certification Data
- Emergency Responder Data
- Cardholder Role
- NACI Indicator

Case: An emergency occurs and an individual's point of contact must be retrieved

Emergency scenarios that may occur to a PIV cardholder can leverage the functionality provided by BAE. If a visiting PIV cardholder has, for example, a medical emergency the identifying information found within the PIV system, coupled with rich attribute exchange can facilitate the retrieval of attributes corresponding to an individual. These attributes must generally be available on the contactless interface to support scenarios where a cardholder is unable to respond.

Variants

- None Determined

Sample Attributes

- Emergency Point of Contact Data

Case: An individual requests a meeting in advance (Interagency Visit Request)

Individual PIV cardholders may exchange attributes necessary for authorization in advance in preparation for a visit to an external agency. Attributes such as the CHUID can be shared in advance to provision local PACS systems and expedite security processing. The attributes exchanged are determined by the relying party and illustrate a culmination of attributes from previous use cases.

Variants

- Subset Case #1: Employee of Agency X remotely presents a PIV card at a location operated Agency Y
- Subset Case #2: Employee of Agency X remotely presents a PIV card at a subsidiary facility of Agency X

Sample Attributes

- CHUID
- Photo
- Clearance
- Certifications
- Emergency Point of Contact Data

Appendix D: Glossary

Term	Definition
Attribute Authority	Entity providing Backend Attributes to the requesting BAE Relying Party.
Authoritative Source	The Authoritative Source for a Backend Attribute is the entity that maintains the attested version of that Backend Attribute. When more than one entity (e.g., another Attribute Authority, a RP) has the same Backend Attribute, the Authoritative Source's value must be considered the correct value, and should take precedent over all other values. Only one Authoritative Source should exist per Backend Attribute.
Backend Attribute Exchange (BAE)	Standard mechanism for Relying Parties to obtain PIV Cardholder information (Backend Attributes) from the Authoritative Source (Attribute Authority).
Backend Attributes	PIV Cardholder information stored by an Attribute Authority available to Relying Parties typically to support PIV Cardholder authentication, authorization, or emergency events.
BAE Broker	The Broker is the communications conduit between RPs and Attribute Authorities.
BAE Certificate	Possession of a valid BAE Certificate is required to sign or encrypt BAE messages, and therefore participate in BAE as a trusted partner.
BAE External Service	Handles the exchange of Backend Attributes between trusted BAE partners.
BAE Internal Service	Handles the exchange of Backend Attribute data between local attribute authorities.
BAE Requester	BAE Broker that sends a request for Backend Attributes.
BAE Responder	BAE Broker that returns Backend Attribute values that were requested by a BAE Requester.
Batch Processing	A data processing operation and where related BAE transactions are grouped together and transmitted for processing in one group.
Cardholder Unique Identifier (CHUID)	The CHUID is defined to provide the basis for interoperable identification of individuals and to extend capabilities over magnetic stripe technology for Physical Access Control System applications. It contains a series of mandatory and optional tagged objects. Some of these include the Federal Agency Smart Credential Number (FASC-N), the Global Unique ID (GUID), and the Asymmetric Signature.
Extensible Markup Language (XML)	Specification developed by the W3C. XML is a pared-down version of SGML, designed especially for Web documents. It allows designers to create their own customized tags, enabling the definition, transmission, validation, and interpretation of data between applications and between organizations.
Federal Agency Smart Crednetial – Number (FASC-N)	The FASC-N is the primary identification string to be used on all government issued credentials.
HyperText Transfer Protocol (HTTP)	Underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands. In the Federation, where appropriate, HTTP is used to redirect end users.

Term	Definition
Metadata	Message exchange between two BAE entities requires each to have specific knowledge about the other. One example is the URL of each entity a BAE Broker technically interoperates. Without such knowledge, a BAE Broker does not know where to send messages for processing. Metadata describes and conveys such information.
National Information Exchange Model (NIEM)	NIEM is a partnership of the U.S. Department of Justice and the Department of Homeland Security. It is designed to develop, disseminate and support enterprise-wide information exchange standards and processes that can enable jurisdictions to effectively share critical information in emergencies, as well as support the day-to-day operations of agencies throughout the nation. NIEM enables information sharing, focusing on information exchanged among organizations as part of their current or intended business practices. The NIEM exchange development methodology results in a common semantic understanding among participating organizations and data formatted in a semantically consistent manner. NIEM will standardize content (actual data exchange standards), provide tools, and managed processes. NIEM builds on the demonstrated success of the Global Justice XML Data Model. Stakeholders from relevant communities work together to define critical exchanges, leveraging the successful work of the Global Justice XML Data Model.
PIV Card	A physical artifact (e.g., identity card, “smart” card) issued to an individual that contains stored identity credentials (e.g., photograph, cryptographic keys, digitized fingerprint representation) so that the claimed identity of the PIV Cardholder can be verified against the stored credentials by another person (human readable and verifiable) or an automated process (computer readable and verifiable).
PIV Card Issuer	An authorized identity card creator that procures FIPS-approved blank identity cards, initializes them with appropriate software and data elements for the requested identity verification and access control application, personalizes the cards with the identity credentials of the authorized subjects, and delivers the personalized cards to the authorized subjects along with appropriate instructions for protection and use.
Relying Party	Entity requesting Backend Attributes typically to support PIV Cardholder authentication, authorization, or emergency events.
Security Assertion Markup Language (SAML)	The set of specifications describing security assertions that are encoded in XML, profiles for attaching the assertions to various protocols and frameworks, the request/response protocol used to obtain the assertions, and bindings of this protocol to various transfer protocols (for example, SOAP and HTTP). SAML addresses web single sign-on, web services authentication, attribute exchange, authorization, non-repudiation, and secure communications. SAML defines assertion message formats that are referenced in Liberty Alliance, Shibboleth, WS-Security, and other specifications. SAML has become the standard web SSO identity management solution. Several versions have been released to date, including SAML 1.0, SAML 1.1, and SAML 2.0. The Organization for the Advancement of Structured Information Standards (OASIS) oversees SAML.

Term	Definition
Service Provisioning Markup Language (SPML)	An XML-based framework, developed by OASIS, for exchanging user, resource and service provisioning information between cooperating organizations. SPML relies on SAML for the exchange of authorization data. Several versions have been released including version 1.0 in 2003 and version 2.0 in 2006.
Shared BAE Broker	A BAE broker used by multiple departments or agencies to participate in Backend Attribute exchanges.
Simple Object Access Protocol (SOAP)	Lightweight XML-based messaging protocol used to encode the information in Web service request and response messages before sending them over a network. It consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined data types, and a convention for representing remote procedure calls and responses. SOAP messages are independent of any operating system or protocol and may be transported using a variety of Internet protocols, including MIME and HTTP.
Trust Partner	An organization's network within the BAE Trust Model.
XPATH	A language for selecting nodes from an XML document. XPath may be used to compute values from the content of an XML document. Current production versions include 1.0 and 2.0.

8 Appendix E: Acronyms

Acronym	Definition
AA	Attribute Authority
AWG	Architecture Working Group
BAE	Backend Attribute Exchange
CDP	Certificate Revocation List Distribution Point
CHUID	Cardholder Unique Identifier
COTS	Commercial off the Shelf
CRL	Certificate Revocation List
FASC-N	Federal Agency Smart Credential - Number
FEMA	Federal Emergency Management Agency
FICC	Federal Identity Credentialing Committee
HSPD-12	Homeland Security Presidential Directive #12
HTTP	HyperText Transfer Protocol
HTTPS	Hypertext Transport Protocol, Secure
ID	Identifier
IT	Information Technology
JPEG	Joint Photographic Experts Group
LACS	Logical Access Control System
NIEM	National Information Exchange Model
NIST	National Institute of Standards and Technology
OASIS	Organization for the Advancement of Structured Information Standards
OCSP	Online Certificate Status Protocol
ORG	Organization
PACS	Physical Access Control System
PIV	Personal Identity Verification
PKI	Public Key Infrastructure
RFC	Request For Comment
RP	Relying Party
SAML	Security Assertion Markup Language
SOAP	Simple Object Access Protocol
SPML	Service Provisioning Markup Language
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform resource Locator
WS	Web Services
XML	Extensible Markup Language