re:ID

Summer 2011

Regarding ID Magazine – a survey of identification technology • SecureIDNews • ContactlessNews • CR80News • RFIDNews

PIV-I

Is it the standard for *all* future IDs?

Private sector use of the FIPS 201 standard could dwarf government use

Web services look to revolutionize biometrics

New protocols could cut cord to biometric readers, end need for software drivers

Jill Jaracz
Contributing Editor, AVISIAN Publications

The National Institute of Standards of Technology (NIST) is working to establish protocols to ease implementation and increase interoperability for biometric devices. A team within NIST sees Web services as the key to extending biometrics across platforms, solutions and devices.

The Biometric Web Services (BWS) project, a five-person team within NIST's Information Technology Laboratory Image Group, is creating specifications for biometric devices to use Web services for interoperability.

Ross J. Micheals, NIST's supervisory computer scientist and leader of this project, explains that in the current climate, sensors and matchers need to be built from the ground up. As new technology comes on the market, the devices may not necessarily be able to interact with a user's current system. In a nutshell, Biometric Web Services is aiming to make it easier to bring biometric capabilities to more devices than ever before – devices that would otherwise require an investment in a specific combination of hardware and software.

Currently biometric devices require dedicated software to interact with other electronic devices (e.g. computers, handhelds, mobiles). When either device changes, the wheel must essentially be recreated – or at least the software that drives it. "If devices can understand the Web inherently and the device changes," explains Micheals, "you don't have to rely on the software that hinders interoperability."

About five years ago, NIST decided to create specifications to determine if biometric devices were viable over Web protocols. The Biometric Web Services project was formed.

The group is focused on the creation of two basic protocols that would eliminate the need for biometric devices to have dedicated connections and dedicated device drivers, says Micheals. If successful, users would be able to control any device from anywhere.

In terms of physical connectivity, the use of Web services will eliminate the need for a USB or IEEE 1394 connection and device drivers. Instead, the connection can be made via Ethernet or Wi-Fi. Additionally, a system won't need to rely on device-specific, software-based drivers.

Web services also change logical connectivity in that devices could be shared from an Internet-enabled device, such as a tablet or handset, and it won't matter whether or not the device operates on the same platform.

"There is no reason why devices shouldn't inherently understand Web protocols. It's very tractable to have this technology in a small handheld form," Micheals says. With the Biometric Web Services protocols, mobile devices can be programmed to talk to the Web and no longer need data storage capabilities. "What we're really trying to do is to describe an outlet, cut the cord on the biometric sensor, and define a clear boundary between components of the system."

The team's first major release was a working demonstration of the project, which it presented at the Biometric Consortium Conference in September 2010.

Since then, Biometric Web Services has worked with the OASIS standards body on an implementation of the OASIS Biometric Identity Assurance Services (BIAS) specification. OASIS' BIAS Integration Technical Committee is determining a standard way to access remotely invoked biometric services via a services-oriented framework.

This effort has resulted in the team's January 2011 release of a simple implementation of the BIAS spec in which a client and server use arbitrary binary data to show BIAS' various functions. In February OASIS released a draft of this specification and is accepting comments and suggestions from the public to aid in its revision.

Challenges to protocol creation

In developing the protocols, the team has run into some challenges. One hurdle has been developing multi-user capabilities. Because biometric sensors are currently built as single-user devices, they will have to be built with concurrent access capabilities in order to incorporate the multiuser functionality of Web services.

The team is also trying to answer questions around live previewing capabilities and multilayered security. "Local door access has different requirements versus logging onto a Web site," Micheals says. "How you secure, encrypt data, and how it travels through the system are important questions you have to think about when designing solutions."

One of the project's sponsors is the Department of Homeland Security, who can use Biometric Web Services with its systems, particularly as these systems and components age. Operating within a closed system, Biometric Web Services can work with a mixture of new and old technology that still has the capability to interact because all the components are designed with the same protocol.

NIST hopes these standards will help drive technology and stimulate the market. "When [devices] all talk the same language, markets open up," Micheals says. When the protocols are established, customers will be able to purchase products that work with any existing Web protocol system, and developers will be able to add their own value to the devices. "There's a potential market for biometric access control devices that are more interchangeable. Certainly as a consumer, that makes it more attractive," Micheals says.

Micheals says that NIST is reaching out to the public for feedback and input. "Our mission is to do the best thing technically, but we know a lot of excellent technology work is done in the private sector, so that's where we're trying to get help," Micheals says.

The Biometric Web Services team is fostering relations with experts in industry, academia and government to get input on the protocols. It maintains a listserv for announcements about projects and discussion from individuals.

The team is being careful to create flexible protocols that can be extended and won't render future technology or modalities incompatible. "We don't know what might be needed to add to the list later. [They should be] not so strict that you couldn't extend them," Micheals says.

Work on these two protocols will continue through the majority of 2011. BWS intends to have a final draft of the specifications written by the end of NIST's fiscal year, which is in October.

ID

Web services defined

Web Services: The Internet standards body W3C defines Web services as "a software system designed to support interoperable machine-to-machine interaction over a network." It uses a machine-processable format such as WSDL and standardized SOAP message formats, HTTP and XML. Web services can implement a Service-oriented architecture (SOA), which is a flexible way to design an ecosystem of interoperable services that work with multiple systems across various domains.

WSDL: Web Services Description Language, an XML-based language used in combination with SOAP to enabled clients programs to find and connect to Web services over the Internet.

SOAP: The Simple Object Access Protocol was designed in 1998 a specific way to exchange structured information used by Web Services. Messages are exchanged in XML using application-layer protocols such as HTTP or RPC (Remote Procedure Call).

HTTP: Hypertext Transfer Protocol, developed since 1991, is a networking program for distributed information systems and forms the basis for communicating over the Web. HTTP/1.1, defined in RFC 2616, defines nine verbs (methods) for manipulating remote resources.

XML: Extensible Markup Language, an open standard for encoding documents in machine-readable form, was published in 1996 with the goals of simplicity and ease of use over the Internet. Since then, most modern APIs and file formats have been developed on top of XML-based formats, such as RSS, Atom, SOAP, XHTML, and various office document formats.

REST: Representational State Transfer, introduced in 2000 is an "architectural style", simpler than SOAP, which defines the interactions between clients and servers (requests and responses) and the data they exchange (resources and representations). It typically a subset of the HTTP verbs (POST, GET, PUT, and DELETE) to implement create, read, update, and delete operations on remotely accessible resources.