ITL BULLETIN FOR JULY 2016

IMPROVING SECURITY AND SOFTWARE MANAGEMENT THROUGH THE USE OF SWID TAGS

David Waltermire, Larry Feldman,[1] and Greg Witte,[1] Editors
Computer Security Division
Information Technology Laboratory
National Institute of Standards and Technology
U.S. Department of Commerce

Background

An important component of maintaining organizational information security is *software asset management* (SAM), especially in the area of *software inventory management*. Software inventory management includes the need to confirm and monitor the extent to which software is installed in accordance with organizational requirements (e.g., change management, licensing requirements, patching regimens, and removal processes).

NIST's Information Technology Laboratory has been working with the security and SAM communities to conduct and publish research regarding the use of software identification (SWID) tags to support software asset management and software security. NIST has also been working with the international standards development organizations to enhance the SWID tag standards and to promote their use in software and security management protocols.

Introduction

The International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) standard 19770-2, *Software asset management -- Part 2: Software identification (SWID) Tags*, specifies the use of SWID tags for a broad range of SAM use cases. NIST worked with the SAM and security community to produce an updated version of the ISO/IEC standard, released in 2015.

ITL's Computer Security Division recently released NIST Internal Report (NISTIR) 8060, *Guidelines for the Creation of Interoperable Software Identification (SWID) Tags*. This NISTIR introduces SWID tags in an operational context, provides guidelines for the creation of interoperable SWID tags, and highlights key usage scenarios for which SWID tags are applicable. In addition to helping introduce SWID tags to a broader audience, the report provides specific tag implementation guidelines that supplement the SWID tag standard, and provides a set of operational usage scenarios that illustrate how SWID tags conforming to these guidelines can be used to achieve a variety of cybersecurity goals. By following the

[1] Larry Feldman and Greg Witte are Guest Researchers from G2, Inc.

guidelines in this report, tag producers can have confidence that they are providing the necessary data, with the requisite data quality, to support operational security goals.

SWID Tag Types and the Software Life Cycle

NISTIR 8060 discusses the various types of tags defined by the SWID specification: *primary*, *patch*, *corpus*, and *supplemental*. Primary, patch, and corpus tags have similar functions in that they describe the existence and/or presence of different types of software, and, potentially, different states of software products. In contrast, supplemental tags furnish additional information not contained in corpus, primary, or patch tags.

Drawing from NISTIR 8060, the four tag types are:

- **Primary** – Identifies and describes software products that have been successfully installed on a computing device, and information about those products;
- **Patch** – Identifies and describes each patch installed on an endpoint. A patch tag can be placed on the device when a patch is installed, or discovery tools can create a patch tag to indicate the previous application of a patch;
- **Corpus** – Identifies and describes products in a pre-installation state, such as in an installation package or media (e.g., an installation CD). Corpus tags support integrity verification of software to be installed; and
- **Supplemental** - Provides a flexible and extensible means to augment the information provided in a corpus, primary, or patch tag to assist with the overall management of software, supporting related processes (e.g., configuration management, vulnerability management).

All four tag types come into play at various points in the software life cycle. The following examples illustrate the use of the four tag types to provide information that is critical in supporting and informing software management life-cycle processes:

- **Software Pre-Installation.** Before a software product is installed, a corpus tag provides hash values that can be used to verify the integrity of the installation files.
- **Software Installation.** A primary tag will be installed with the software product (or subsequently created) to uniquely identify and describe the product. Supplemental tags are created to augment primary tags with additional site-specific or extended information. Patch tags may also be installed during software installation to provide information about software fixes deployed along with the base software installation.
- **Software Patching.** When a new patch is applied to the software product, a new patch tag is installed, supplying details about the patch and its dependencies.

- **Software Upgrading.** As a software product is upgraded to a new version, new primary and supplemental tags replace existing tags for the older software version, enabling timely and accurate tracking of updates to software inventory.
- **Software Removal.** Upon removal of the software product, relevant tags are removed reflecting the product's removal.

Installation, patching, upgrading, and removal events can trigger timely reporting of changes to a devices software inventory using the identification information in a tag.

Guidelines for the Creation of Interoperable SWID Tags

By deploying tags in consistent locations, and through the use of standardized attributes in a standardized XML format, SWID tags enable automated tools to reliably and accurately determine and maintain software inventory. To maximize interoperability, NIST has provided a number of guidelines that extend ISO/IEC 19770-2, each with a unique coded identifier that helps group the guidelines into categories. These guidelines help to support:

- Creation and maintenance of SWID tags that are well-formed (as described in the ISO 19770-2:2015 standard) and reliable for use by discovery tools;
- International use of SWID tags by providing language-dependent attribute values in region-specific human languages;
- Consistent methods to provide detailed information about the files composing an installed software product. This information helps when measuring that the correct files are installed and have not been modified by unauthorized activities; and
- The use of accessible repositories to make tag information available for use by other management and cybersecurity processes.

Software Management as Part of Managing Cybersecurity

The use of SWID tags supports a broad array of cybersecurity use cases, especially in the use of automated software discovery or monitoring tools (referred to generically in the report as "discovery tools"). Because the tags are created by software vendors in conformance with an international standard, they are highly interoperable and reliable.

Historically, software discovery tools have had to rely upon proprietary methods for identifying installed software products on an organization's endpoints. Discovery tools have had varying levels of success in helping an organization reliably understand what authorized and non-authorized software existed in the enterprise. SWID tags help these tools to consistently recognize installed software, supporting the continuous monitoring of software inventory changes.

SWID tags also help organizations to track the software life cycle for custom-built and/or in-house developed software, further supporting automated continuous monitoring for an enterprise. While many discovery tools are able to recognize some well-known commercial and government off-the-shelf (COTS and GOTS) software applications, they may not have the intelligence to recognize all custom software products. Through the use of SWID tags created for custom software, monitoring of custom software can be performed using the same mechanisms as commercially available or open-source software. This helps organizations to incorporate management of custom software products as part of an integrated continuous monitoring solution.

SWID Tag Usage Scenarios

NISTIR 8060 presents a number of usage scenarios illustrating how SWID tags, created in alignment with the report's guidelines, help to improve the security of endpoints on enterprise networks.

The first set of usage scenarios illustrates how SWID tags help an organization to maintain awareness of vulnerabilities related to installed software, including the state of patches applied to endpoints. They also show how SWID tags aid in the automated correlation of information published by vulnerability information sources (e.g., NIST's National Vulnerability Database, vulnerability advisories issued by vendors and independent security analysts) with the inventory information collected by discovery tools.

A second set of usage scenarios focuses on the use of SWID tags to help security practitioners minimize security risks by enforcing enterprise policies regarding authorized software. These policies may be implemented as blacklists (lists of prohibited products, with all unlisted products implicitly allowed) or whitelists (lists of allowed products, with all others prohibited). In addition, specific products may be designated as mandatory by the enterprise (e.g., antivirus and intrusion detection and prevention applications), possibly based upon the endpoint's role (e.g., end-user workstation, Internet-facing web server).

A final usage scenario considers a forward-looking approach to improving an organization's cybersecurity by preventing potentially vulnerable endpoints from connecting to the network, or to move such endpoints to an isolated network segment for remediation or investigation. Currently, products are available that achieve this through proprietary methods and groups are working on open standards to accomplish this goal. For example, the Trusted Computing Group's Trusted Network Communications Working Group (TNC-WG) has defined an open solution architecture that enables network operators to enforce policies regarding the security state of endpoints in order to determine whether to grant a connecting device access to a requested network infrastructure. The use of SWID tags provides a technology-neutral way to verify an endpoint's compliance with certain configuration policies (e.g., updated antivirus definitions, configuration compliance with baseline specifications) and safeguards against known software vulnerabilities (e.g., missing patches).

Role of SWID Tags in Security Automation

Because SWID tags help to provide consistent and accurate software inventory information, the use of SWID tags as installation evidence is described in the next minor version of the Security Content Automation Protocol (SCAP), SCAP 1.3 (currently in development). SWID tags are also part of the ongoing work in the Security Automation and Continuous Monitoring (SACM) working group of the Internet Engineering Task Force (IETF). Information about the SACM work is available from https://datatracker.ietf.org/wg/sacm/charter. Interested parties may subscribe to the SACM mailing list at https://www.ietf.org/mailman/listinfo/sacm. The international standards being developed in the IETF SACM working group are expected to be the basis for the next major version of SCAP, SCAP 2.0.

Conclusion

SWID tags offer many benefits to software providers, software consumers, and providers of inventory-based tools and services by providing improved capabilities to manage enterprise software inventory information, using consistent, interoperable, and accurate data. Broad adoption of SWID tags by software providers and support in security tools is key to success. NISTIR 8060 provides an important first step—promoting awareness, providing implementation guidance, and motivating usage scenarios. NIST encourages adoption of SWID tags to support software inventory, participation in standards activities incorporating the use of SWID tags, and in the ongoing use of SWID tags to support information security activities.

ITL Bulletin Publisher: Elizabeth B. Lennon
Information Technology Laboratory
National Institute of Standards and Technology
elizabeth.lennon@nist.gov