ivanti

Ivanti Content Research, Testing and Validation of Authenticity

White Paper

Contents

This document is provided strictly as a guide. No guarantees can be provided or expected. This document contains the confidential information and/or proprietary property of Ivanti, Inc. and its affiliates (referred to collectively as “Ivanti”) and may not be disclosed or copied without prior written consent of Ivanti.

Ivanti retains the right to make changes to this document or related product specifications and descriptions, at any time, without notice. Ivanti makes no warranty for the use of this document and assumes no responsibility for any errors that can appear in the document nor does it make a commitment to update the information contained herein. For the most current product information, please visit www.Ivanti.com.

Copyright © 2020, Ivanti. All rights reserved

1. Ivanti Content Research

1.1. Ivanti Content Team

Ivanti's Content team is made up of Analysts, QA, and Developers. Analysts actively create content and build the detection and deployment logic. QAs validate each content build against a variety of test scenarios. Developers extend Ivanti's content system to add new metadata, scrape content (e.g. Microsoft KBs, CVEs, severity, etc), setup sensors to detect when new updates become available, extract installers to determine what they are designed to do and autogenerate content.

1.2. Ivanti Content Environments

Ivanti runs a combination of VMware and Azure virtual infrastructures with hundreds of VMs with many different states that can be called up on demand. Ivanti has a home-grown test and automation framework that can be defined with many sets of test systems so when something needs to be tested, like SQL or .Net Framework, Ivanti can bring the correct set of systems online and start testing very quickly. Ivanti keeps a variety of states as well since no two environments are alike. We keep a set of systems that are up to the latest available for each product, some that are in a middle state like 3 to 4 releases behind, and a set of completely unpatched systems. This allows us to test a variety of permutations that often flushes out detection discrepancies based on current state of the system.

1.3. Content Generation

A variety of steps are run when a new update comes out. Ivanti has sensors that detect a new update is available. Like in the case of Adobe, Ivanti often picks the update installer 4 to 6 hours before a KB article goes online. The new update is downloaded, signatures are validated, the installer is extracted, and a set of automation tools called producers are run to setup an initial set of detection and deployment logic. This gets into queue and an Ivanti analyst completes a human verification of the content to determine if anything is missing or out of the ordinary. Next this is put into a build. Ivanti's content team builds the pass into a testing queue where one of the content QA team runs the update through Ivanti's test framework. This runs the update against all possible operating systems it can be installed on with a variety of states to determine if the update meets detection and install criteria. If the update passes, it goes on to the production build. If it fails, it is returned to an analyst with the detection or install bugs that need to be resolved.

2. Testing and Validation

2.1. New Product Research and Testing

When Ivanti researches a new product, the company validates download centers, authenticity of the packages, finds any documentation on detection and install switches, etc. Next, Ivanti cracks open the installer and evaluates what it is going to do to a system to check what files are updated, file versions, registry keys and so on. Once Ivanti determines that an update meets the criteria of being a single installable package, with detectable validation like files and registry keys, install switches to support silent automated installation and can disable reboot, it is added to the catalog.

2.2. Content Regression Testing and Validation

Because there are so many variables and ways that things could go wrong with patching, Ivanti has developed a variety of additional validation steps to check work. The content team checks against other sources like the Windows Catalog. More often this flushes out ways that Ivanti may catch a patch that Microsoft's Windows Update or WSUS detection logic (both different by the way) did not catch. On occasion, the content team finds things that went undocumented such as the update being applicable to another edition of an OS rather than what the documentation stated. The content team runs download validation that spot checks the installers to make sure they are still available and that they have not been changed. Vendors have been known to sneak an updated binary into the mix without notifying anyone it has happened. There is also the common straggler that needs to be picked up.

Often Microsoft will release an update on Patch Tuesday and by Thursday there will be one or two new variations that release, either an updated binary or an additional edition or OS that has been covered. For most of Ivanti's competitors, these types of changes go unnoticed for some time, but Ivanti tends to flush them out quickly. Because the test environment and content are highly automated, Ivanti can reply any content release. The content team will reply Patch Tuesday testing a few content releases later to see if any precedence issues are flushed out (cases where a new update flushes out or modifies how to validate an older update).

2.3. Validation of Authenticity for Patches

Ivanti has several solutions that can deliver updates and each of those solutions may have different means of download and distribution within a customer's internal environment. To ensure authenticity of an update Ivanti performs the following activities. During testing, Ivanti validates the authenticity of the download centers for each vendor, validates digital signatures, and MD5 hashes. Depending on the Ivanti solution, digital signatures are always validated and for EPM Patch and Patch for SCCM the hashes are also evaluated upon download to the customers onsite repository. Additional validation is performed for EPM Patch and Security Controls and Patch for SCCM as the updates are distributed to repositories and executed on endpoints to prevent tampering with the updates ensuring delivery of the authentic patch from the vendor.

3. Crowdsourcing

Ivanti has a rather large customer base and many OEM vendors use Ivanti's technology. This means Ivanti catches a lot of edge cases that other vendors do not. Edge cases can occur in a number of ways. Often, they are a result of environment variables. If there is a 3rd party or custom product that received custom binaries for a Microsoft component like GDI+ for rendering image software in their medical records system, the versions on those custom binaries could mess with detection because they are not a publicly documented version. Ivanti has had cases where a patch released more than a year prior came up with an edge case that. In these scenarios, Ivanti worked with the customer and found a way to detect properly. If the customer tried to scan with Microsoft WU, WSUS\SCCM, MBSA, or even other competitors, that edge case was undetectable.

Another way Ivanti catches these cases is from the vulnerability vendors. Vulnerability vendors often pick up ways to detect a vulnerability and then the customer reaches out to us. Ivanti works through detection logic to properly detect the edge case and similarly it is undetectable with other patch vendors. These edge cases often affect a very small subset of customers because of unique variables in their environments.

For more information on Ivanti's services and products, please visit **www.ivanti.com.**

www.ivanti.com

1.800.982.2130

sales@ivanti.com