

**IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF COLORADO**

**Civil Action No. 1:22-cv-00643**

**IVANTI, INC.,**

**Plaintiff,**

**Jury Trial Demanded**

**v.**

**PATCH MY PC, LLC,**

**Defendant.**

**COMPLAINT FOR PATENT INFRINGEMENT**

Plaintiff Ivanti, Inc. (“Ivanti” or “Plaintiff”) hereby submits this Complaint for patent infringement against Defendant Patch My PC, LLC (“Patch My PC” or “Defendant”) and states as follows:

### **THE PARTIES**

1. Plaintiff Ivanti, Inc. is a corporation existing under the laws of Delaware with its principal place of business at 10377 South Jordan Gateway Suite 110 South Jordan, Utah 84095.

2. Ivanti is informed and believes, and on that basis alleges, that Defendant Patch My PC, LLC is a Colorado limited liability company with its principal place of business at 858 W Happy Canyon Rd, Suite 260 Castle Rock, CO, 80108. Patch My PC may be served by serving its registered agent, Justin Chalfant, at 858 W. Happy Canyon Rd, Suite 260, Castle Rock, CO, 80108.

### **NATURE OF THE ACTION**

3. This is a civil action for infringement of U.S. Patent Nos. 6,990,660 (“the ’660 Patent”); 7,823,147 (“the ’147 Patent”); and 8,407,687 (“the ’687 Patent”), arising under the patent laws of the United States, 35 U.S.C. § 1 *et seq.*

### **JURISDICTION AND VENUE**

4. This Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a) because this action arises under the patent laws of the United States, 35 U.S.C. §§101 *et seq.*

5. The Court has general personal jurisdiction over Defendant Patch My PC because Defendant Patch My PC’s principal place of business is within the District of Colorado. Further, Patch My PC is incorporated within the District of Colorado.

6. Venue is proper in this federal district as to Patch My PC pursuant to 28 U.S.C. §§ 1391(b)-(c) and 1400(b). Patch My PC resides in the District of Colorado; it is both incorporated in Colorado and has its principal place of business in Colorado.

**PATENTS-IN-SUIT**

7. The patents-in-suit relate to systems and methods for updating existing software across a remote network based on the use of patch fingerprints to check for the need to update software and then updating that software as required. Each of the asserted patents claim priority to provisional U.S. Patent Application No. 60/234,680 entitled “Non-invasive Automatic Offsite Updating System and Method,” filed in the United States Patent and Trademark Office (USPTO) on September 22, 2000.

8. On January 24, 2006, the USPTO duly and legally issued U.S. Patent No. 6,990,660 entitled “Non-invasive Automatic Offsite Patch Fingerprinting and Updating System and Method.” Ivanti owns all substantial rights to the ’660 Patent, including the right to sue and recover damages for all infringement thereof. Attached hereto as Exhibit A is a true and correct copy of the ’660 Patent.

9. On October 26, 2010, the USPTO duly and legally issued U.S. Patent No. 7,823,147 entitled “Non-invasive Automatic Offsite Patch Fingerprinting and Updating System and Method.” Ivanti owns all substantial rights to the ’147 Patent, including the right to sue and recover damages for all infringement thereof. Attached hereto as Exhibit B is a true and correct copy of the ’147 Patent.

10. On March 26, 2013, the USPTO duly and legally issued U.S. Patent No. 8,407,687 entitled “Non-invasive Automatic Offsite Patch Fingerprinting and Updating System and

Method.” Ivanti owns all substantial rights to the ’687 Patent, including the right to sue and recover damages for all infringement thereof. Attached hereto as Exhibit C is a true and correct copy of the ’687 Patent.

### **BACKGROUND**

11. The patents-in-suit relate to “[m]ethods, systems, and configured storage media . . . for discovering software updates, discovering if a given computer can use the software update, and then updating the computers with the software as needed automatically across a network without storing the updates on an intermediate machine within the network.” ’660 Patent at Abstract. If the update fails, then the target computer may be restored to a non-update state. *See id.* at 3:26-31. The inventions “facilitate[ ] software deployment, software installation, software updating, and file distribution based on software and patch finger printing across multiple operating systems and devices, across a network.” *Id.* at 3:33-37.

12. Figure 2 of the ’660 Patent illustrates an embodiment of the patented inventions:

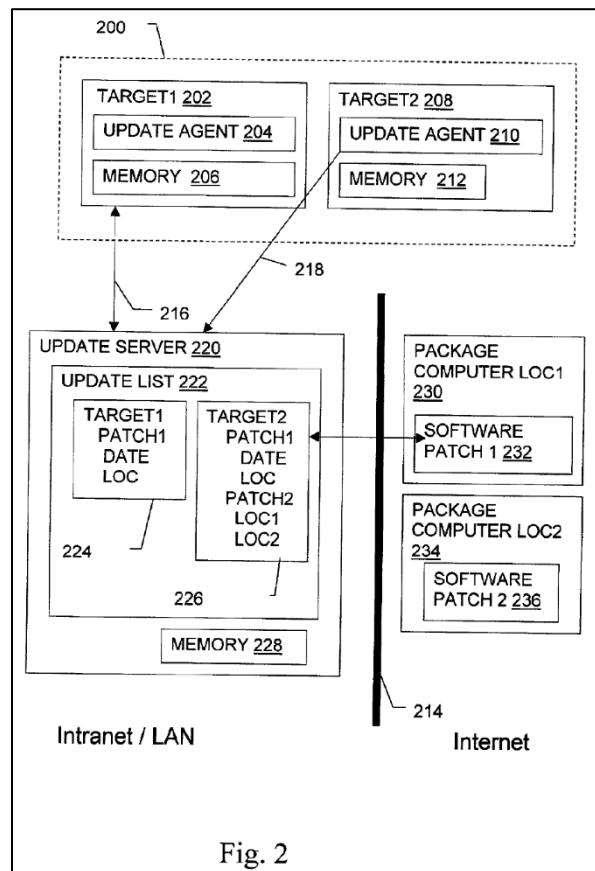


Fig. 2

'660 Patent at Fig. 2. As illustrated, the systems and methods of the patents-in-suit involve a package computer, an update server, and a target computer. The package computer maintains the software patches that may be needed to update target computers within the network. *See id.* at 3:44-48. The update server accesses these software patches so that they can be deployed to the target computers within the network as needed. *See id.* at 4:30-54. The update server may include a repository component that maintains patch fingerprint information as well as inventory information about the target computers in the network. *See id.* at 3:56-66.

13. Software patch information is maintained in patch fingerprints, allowing the system “to determine if a given software package (associated with the patch fingerprint), patch driver, etc. should be loaded onto a [target] computer in the system.” *Id.* at 3:56-59. “Using the

information in the patch fingerprint, the inventory library, and specific information gleaned from each network target computer, the system is able to intelligently recommend which patches and drivers are required for a given computer.” *Id.* at 4:66-5:3. The patch fingerprint includes various information defining a software patch or update, such as an existence test, a signature block, and/or install info. *See id.* at Fig. 9 (patch fingerprint 906).

14. The system and methods of the invention may also include a discovery agent for use on the target computers. This discovery agent can be used to “discover[ ] the hardware and software on [a target] machine” and to “return scan results for patch fingerprints, which indicate whether it is appropriate to install a specific patch associated with each patch fingerprint.” *Id.* at 4:4-14.

**CLAIM 1**  
**INFRINGEMENT OF U.S. PATENT NO. 6,990,660 BY PATCH MY PC**

15. Patch My PC has infringed directly and continues to infringe directly, either literally or under the doctrine of equivalents, at least claim 1 of the ’660 Patent by its manufacture, sale, offer for sale, and use of any one or more of the patching services offered by Patch My PC including, but not limited to, Enterprise Plus, Enterprise, and Intune only subscriptions. (“the Accused Instrumentalities”). Defendant is therefore liable for infringement of the ’660 Patent pursuant to 35 U.S.C. § 271.

16. Patch My PC received notice of its infringement of the ’660 Patent by letter on November 2, 2021. As of the time Patch My PC first had notice of Ivanti’s allegations of infringement of one or more claims of the ’660 Patent by Defendant, which is no later than November 2, 2021, Defendant indirectly infringed and continues to indirectly infringe at least

claim 1 of the '660 Patent by active inducement under 35 U.S.C. § 271(b). Defendant has induced, caused, urged, encouraged, aided and abetted its direct and indirect customers to make, use, sell, offer for sale and/or import one or more of the Accused Instrumentalities, and thus indirectly infringes at least claim 1 of the '660 Patent. Defendant has done so by acts including but not limited to (1) selling such products including features that—when used or resold—infringe, either literally or under the doctrine of equivalents, the '660 Patent; (2) marketing the infringing capabilities of such products; and (3) providing instructions, technical support, and other support and encouragement for the use of such products, including at least the documents referenced above. Such conduct by Patch My PC was intended to and actually did result in direct infringement by its direct and indirect customers, including the making, using, selling, offering for sale and/or importation of the Accused Instrumentalities in the United States.

17. As a direct and proximate result of Patch My PC's infringement of the '660 Patent, Ivanti has been, is being, and will be damaged. Patch My PC's continued infringement of Ivanti's exclusive rights under the '660 Patent will continue to damage Ivanti, causing irreparable harm for which there is no adequate remedy at law, unless enjoined by this Court.

18. Ivanti is also entitled to recover from Patch My PC the damages sustained by Ivanti as a result of Patch My PC's wrongful acts in an amount subject to proof at trial.

19. As of the time Patch My PC first had notice of the '660 Patent, at least as early as November 2, 2021, Patch My PC has continued with its infringement despite the objectively high likelihood that its actions constitute infringement and Patch My PC's subjective knowledge of this obvious risk. As Patch My PC has no good faith belief that it does not infringe the '660 Patent, at least Patch My PC's continued infringement of the '660 Patent is willful and deliberate, entitling

Ivanti to increased damages under 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

20. For example, the Accused Instrumentalities practice and/or are capable of practicing representative claim 1 of the '660 Patent. The following paragraphs provide details regarding only one example of Patch My PC's infringement, and only as to a single patent claim. Ivanti reserves its right to provide greater detail and scope throughout the discovery process.

21. Claim 1 of the '660 Patent recites:

1. An automated method for at least attempting to update software in a system having a first target computer in a non-update state connected across a network to an update server in a pre-update state, the system also having a package computer which is inaccessible to the first target computer but accessible to the update server, and a repository component accessible to the first target computer and the update server, the method comprising the steps of:
  - (a) putting at least one patch fingerprint which defines a specific software update into the repository component, the patch fingerprint comprising:  
a patch signature and an existence test, wherein the patch signature is configured to request target computer information from the first target computer, and wherein the existence test is configured to use the target computer information provided via the patch signature to determine whether the specific software update is needed on the first target computer;  
wherein the repository component is at least located at the update server and includes recommended configuration information for the first target computer, and
  - (b) gathering the target computer information about the first target computer via a discovery agent located on the first target computer;  
wherein the discovery agent utilizes the patch signature to gather the target computer information,  
wherein the target computer information includes at least hardware configuration information, registry information, software presence information, and software version information relative to the first target computer,  
wherein the target computer information defines current configuration information of the first target computer,
  - (c) sending the target computer information back to the repository component located on the update server,
  - (d) storing the target computer information in the repository component located on the update server,
  - (e) comparing, at the update server, at least a portion of the gathered target computer



- information with the patch fingerprint using the existence test to determine whether the recommended configuration information of the first target computer matches the current configuration information of the first target computer and to determine whether the specific software update is absent from the first target computer and whether the specific software update has a dependency on at least one of another specific software update, a specific software, and a specific hardware;
- (f) if a known condition is met, then placing at least one task identifier on an update task list, the task identifier specifying the first target computer, the update task list stored at the update server, the task identifier also specifying at least one download address which references a location on the package computer that contains a software update for the first target computer;
  - (g) starting a task in response to the task identifier, the task attempting a first download of the specific software update from the package computer to the update server,
  - (h) if the first download completes successfully, then attempting a second download of the specific software update from the update server to the first target computer, wherein during the attempting a second download step, the first target computer is inaccessible to the package computer via a firewall; and
  - (i) monitoring the attempted downloads for an outcome.

U.S. Patent No. 6,990,660 C1 at 1:24-2:27 (Ex Parte Reexamination Certificate 6,819).

22. The Accused Instrumentalities automate updating software for a system of client computers, or target computers. They include a server-side publishing application that relies on cloud-based Patch My PC subscription feeds, which disseminate information about third-party patches available for download and installation. The Accused Instrumentalities also integrate with Microsoft products including Microsoft Endpoint Manager, System Center Configuration Manager (SCCM) and Windows Server Update Services (WSUS), for use as an update server and repository component.

23. The Accused Instrumentalities download software updates from a package computer to an update server. The Accused Instrumentalities also use a patch fingerprint that contains information to identify software updates and their applicability to a target computer. The Accused Instrumentalities gather information about the target computer via discovery agents to

determine the need for an update on the target computers. If an update is needed, it is deployed to the target computer.

24. The Accused Instrumentalities automate updating software for a system of client computers, or target computers. They include a server-side publishing application that relies on cloud-based Patch My PC subscription feeds that disseminate information about third-party patches available for download and installation.

25. The Accused Instrumentalities enhance the operation of Microsoft Endpoint Manager (MEM) which includes System Center Configuration Manager (SCCM) and Intune.

## Simplified Third-Party Application Management

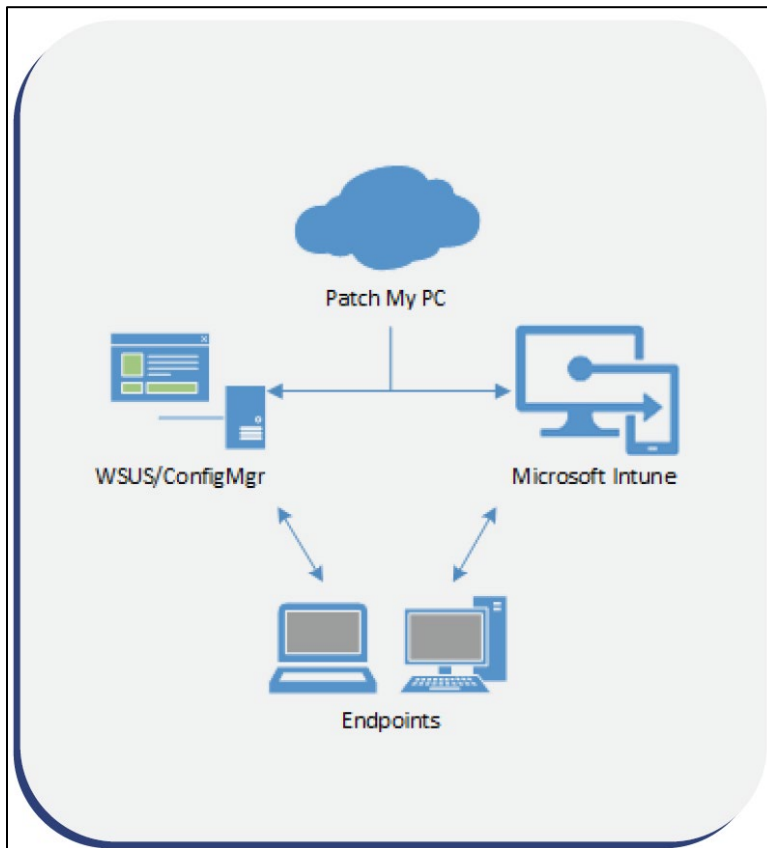
We help you extend Microsoft Endpoint Manager (ConfigMgr and Intune) capabilities by seamlessly integrating third-party patching and application management. Our average

Patch My PC Data Sheet at 1. The Accused Instrumentalities interoperate with MEM, ConfigMgr, Intune, and Windows Server Update Services (WSUS):

This program ensures updates and applications are successfully publishing to WSUS, ConfigMgr, or Intune. When we receive insights that critical operations have failed, we will **notify you proactively about any issues**. We want to ensure our automation is integrated into your day-to-day workflows and just works, so you don't have to worry.

Patch My PC Data Sheet at 2.

26. A high-level architecture diagram for Patch My PC is illustrated below:



Patch My PC Data Sheet at 1. The Patch My PC subscriptions are provided via the Patch My PC cloud, and these are incorporated into WSUS/ConfigMgr via the Patch My PC Publisher application, which is downloaded from Patch My PC’s website:

 **Download the Publisher MSI Installer**



<https://patchmypc.com/msi>

[Download Patch My PC Publisher](#)

<https://docs.patchmypc.com/>.

27. Patch My PC provides a third-party software update catalog that can be directly enabled within Microsoft Configuration Manager.

## Enable third-party updates

Article • 09/17/2021 • 16 minutes to read •  [Is this page helpful?](#) 

*Applies to:* Configuration Manager (current branch)

The **Third-Party Software Update Catalogs** node in the Configuration Manager console allows you to subscribe to third-party catalogs, publish their updates to your software update point (SUP), and then deploy them to clients.

<https://docs.microsoft.com/en-us/mem/configmgr/sum/deploy-use/third-party-software-updates>.

See also <https://docs.microsoft.com/en-us/mem/configmgr/sum/deploy-use/third-party-software-update-catalogs> (listing Patch My PC as a third-party software update catalog provider)

28. Once the catalog is registered in Configuration Manager, metadata describing the available updates will be downloaded to the Windows Server Update Services repository for potential deployment to client computers.

## Subscribe to a third-party catalog and sync updates

When you subscribe to a third-party catalog in the Configuration Manager console, the metadata for every update in the catalog are synced into the WSUS servers for your SUPs. The sync of the metadata allows the clients to determine if any of the updates are applicable. Perform the following steps for each third-party catalog to which you want to subscribe:

<https://docs.microsoft.com/en-us/mem/configmgr/sum/deploy-use/third-party-software-updates>.

29. The Administrator can select which updates should be deployed within the organization. This will cause the update binaries to be added to the WSUS repository.

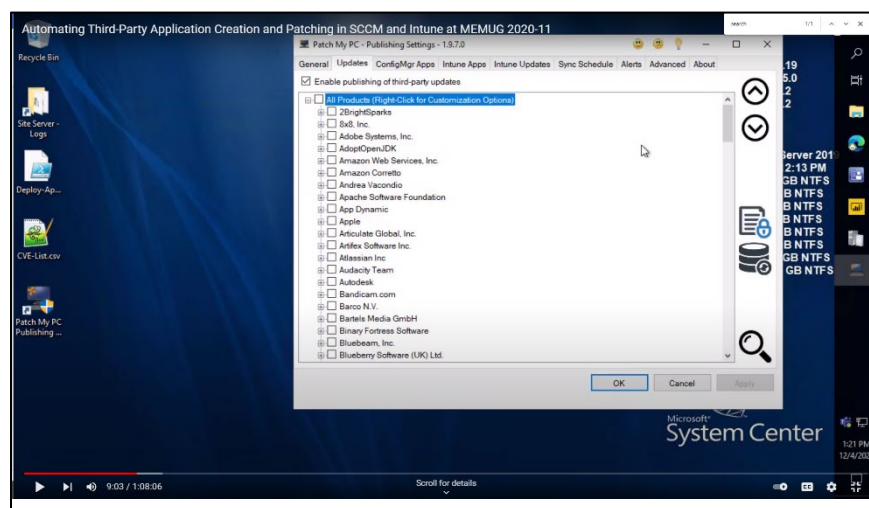
## Publish and deploy third-party software updates

Once the third-party updates are in the **All Updates** node, you can choose which updates should be published for deployment. When you publish an update, the binary files are downloaded from the vendor and placed into the `wsuscontent` directory on the top-level SUP.

<https://docs.microsoft.com/en-us/mem/configmgr/sum/deploy-use/third-party-software-updates>.

A package computer provides the update binaries to update servers which comprise ConfigMgr, Intune, and/or WSUS servers at the direction of Patch My PC's publisher.

30. The Patch My PC tool contains a tab labeled "Updates," which lists the products for which third-party updates are available.



<https://www.youtube.com/watch?v=mp3tQ6Gvqms&t=39s> at 9:03.

31. Patch fingerprints comprise metadata provided about available software updates, and are provided by Patch My PC to update servers and repository components. Patch My PC's catalog includes patch fingerprint data in the form of Software Distribution Packages. A portion of a sample fingerprint is illustrated below for the Zoom application from Patch My PC's limited catalog (available at <https://patchmypc.com/trial-catalog-download>):

```

<sdp:SoftwareDistributionPackage xmlns="http://www.w3.org/2001/XMLSchema"
  <sdp:Properties PackageID="cf4a7cef-0f8e-4a1f-99ac-55fad653530b" Creatio
    <sdp:MoreInfoUrl>https://support.zoom.us/hc/en-us/articles/201361953-N
    <sdp:SupportUrl>https://support.zoom.us/hc/en-us</sdp:SupportUrl>
    <sdp:ProductName>SCUP Updates</sdp:ProductName>
  </sdp:Properties>
  <sdp:LocalizedProperties>
    <sdp:Language>en</sdp:Language>
    <sdp>Title>Zoom Meetings 5.9.3169 (x86)</sdp>Title>
    <sdp>Description>This release contains new features, bug fixes, enhanc
  </sdp:LocalizedProperties>
  <sdp:UpdateSpecificData MsrcSeverity="Important" UpdateClassification="
    <sdp:SecurityBulletinID>PMPC-2022-01-25</sdp:SecurityBulletinID>
    <sdp:KBArticleID>PMPC-2022-01-25</sdp:KBArticleID>
  </sdp:UpdateSpecificData>
  <sdp:SupersededPackages>
    <sdp:PackageID>cf4b3386-9af0-4f01-a13b-b79723e5abdd</sdp:PackageID>
    <sdp:PackageID>040d4d26-cce8-4a19-a508-701709a8f3ce</sdp:PackageID>
    <sdp:PackageID>e469a22f-07f0-46d0-b54b-7d07ea147972</sdp:PackageID>
    <sdp:PackageID>b1a4ebca-c64a-40d0-932c-bb66fca50ae5</sdp:PackageID>
    <sdp:PackageID>c890fc67-4365-4e15-8082-e2fbb52b0dde</sdp:PackageID>
    <sdp:PackageID>3b12b301-395a-401c-9163-419a4486c4c3</sdp:PackageID>
    <sdp:PackageID>9ec77f12-f5eb-49a4-8ddf-b8e4f263f82c</sdp:PackageID>
    <sdp:PackageID>8ecbe740-a995-4272-ac2b-a0f358a4f2b2</sdp:PackageID>
    <sdp:PackageID>e8f8d63a-0cb4-4903-bfbc-f7f2fc9fd37e</sdp:PackageID>
    <sdp:PackageID>dfc308ee-ec9b-40ff-85cb-05c6e7b402f2</sdp:PackageID>
  </sdp:SupersededPackages>
  <sdp:InstallableItem ID="305f4044-7728-4949-a3f5-5537955223e8" >
    <sdp:ApplicabilityRules>
      <sdp:IsInstalled>
        <!-->
      </sdp:IsInstalled>
    </sdp:ApplicabilityRules>
  </sdp:InstallableItem>

```

Trial Catalog Excerpt from cf4a7cef-0f8e-4a1f-99ac-55fad653530b.sdp. The Software Distribution Package is a data structure supported by WSUS for defining software distributions. See [https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb530824\(v=vs.85\)](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb530824(v=vs.85)) (defining the SoftwareDistributionPackage class).

32. The patch fingerprint information requests targets computer information from a first target computer. For instance, the IsInstallable component of the patch fingerprint determines whether the distribution package is installable on the target system based on information gathered from that target. This component is part of the Update Applicability Rules

for a given Software Distribution Package. See [https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902473\(v=vs.85\)](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902473(v=vs.85)) (describing update applicability rules). An example IsInstallable component for Zoom is illustrated below:

```
<sdp:IsInstallable>
  <bar:RegKeyLoop RegType32="true" Key="HKEY_LOCAL_MACHINE"
    Subkey="SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall" TrueIf="Any">
    <lar:And>
      <bar:RegSzToVersion RegType32="true" Key="HKEY_LOOP_TARGET" Subkey=""
        Comparison="LessThan" Data="5.9.3169.0" Value="DisplayVersion" />
      <bar:RegSz RegType32="true" Key="HKEY_LOOP_TARGET" Subkey="" Comparison="EqualTo"
        Data="Zoom" Value="DisplayName" />
      <bar:RegDword RegType32="true" Key="HKEY_LOOP_TARGET" Subkey="" Comparison="EqualTo"
        Data="1" Value="WindowsInstaller" />
    </lar:And>
  </bar:RegKeyLoop>
</sdp:IsInstallable>
```

Trial Catalog Excerpt from cf4a7cef-0f8e-4a1f-99ac-55fad653530b.sdp.

33. The Applicability Rules include Detection methods that query the target computer registry and file system for information. The available rules are defined in the applicability rule schemas.

Detection methods are the different methods defined in the update schemas for use in the applicability rules. Different detection tools expose different detection methods. The most available detection methods are those that query the system registry or file system.

## Applicability rule schemas

Detection methods are defined by one or more of the following schemas:

BaseApplicabilityRules.xsd defines the basic applicability rules, including checks for specific Windows versions and file or registry key existence.

LogicalApplicabilityRules.xsd defines the logical operators (And, Or, Not, True, and False) that allow more complex expressions to be built from the basic rules.

MsiApplicabilityRules.xsd defines the applicability rules used by Windows Installer files.

WindowsDriver.xsd defines the applicability rules used by Windows drivers.

[https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481\(v=vs.85\)](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481(v=vs.85)).

34. The detection methods include gathering information from the target system

registry:

### Registry-based

Registry-based detection methods are easy to use. However, there is no way to guarantee that the state of the registry corresponds to the state of the product or update. For example, a registry key may not be removed when the application is removed, or it may not be updated when a new version is installed.

#### Note

All registry keys used by detection methods must be written under HKEY\_LOCAL\_MACHINE.

The following are some examples of registry-based detection methods (from BaseApplicabilityRules.xsd). For more information on these methods, see [BaseApplicabilityRules Schema](#).

**RegKeyExists** checks whether the specified registry key exists.

**RegValueExists** checks whether a certain registry type and/or value exists for a specified registry key.

**RegDword** compares a registry value to a certain number.

**RegExpandSz** compares a REG\_EXPAND\_SZ value to a certain string.

**RegSz** compares a REG\_SZ value to a certain string.

**RegSzToVersion** compares a REG\_SZ value to a version string.

**RegKeyLoop** evaluates a rule against every subkey of a given registry key.

[https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481\(v=vs.85\)](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481(v=vs.85)).

35. The detection methods also gather information from the target system's file

system:



### File system-based

File system-based detection methods are the most robust kind of detection methods, although they can become extremely complex. For example, an applicability rule that checks the versions of all the files in a product service pack would be a very long rule.

File-system based detection methods can be combined with registry-based methods. For example, an applicability rule can determine whether a patch is installed by checking for its registry value, and confirm the installation state by checking a subset of file versions.

[https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481\(v=vs.85\)](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/bb902481(v=vs.85)).

36. The Applicability Rules detailed above also include existence tests that use gathered information to determine whether the software distribution package is needed on the target computer. For instance, the IsInstalled rule determines whether the package already exists on the target system:

```
<sdp:IsInstalled>
  <bar:RegKeyLoop RegType32="true" Key="HKEY_LOCAL_MACHINE"
    Subkey="SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall" TrueIf="Any">
    <lar:And>
      <bar:RegSzToVersion RegType32="true" Key="HKEY_LOOP_TARGET" Subkey=""
        Comparison="GreaterThanOrEqualTo" Data="5.9.3169.0" Value="DisplayVersion" />
      <bar:RegSz RegType32="true" Key="HKEY_LOOP_TARGET" Subkey="" Comparison="EqualTo"
        Data="Zoom" Value="DisplayName" />
      <bar:RegDword RegType32="true" Key="HKEY_LOOP_TARGET" Subkey="" Comparison="EqualTo"
        Data="1" Value="WindowsInstaller" />
    </lar:And>
  </bar:RegKeyLoop>
</sdp:IsInstalled>
```

Trial Catalog Excerpt from cf4a7cef-0f8e-4a1f-99ac-55fad653530b.sdp.

37. Each software distribution package stored at the repository component comprises recommended configuration information for the target computers. For example, the severity and classification of a software distribution package affects whether it should be installed on target computer systems.

```
</sdp:LocalizedProperties>
<sdp:UpdateSpecificData MsrcSeverity="Important" UpdateClassification="Security Updates" >
  <sdp:SecurityBulletinID>PMPC-2022-01-25</sdp:SecurityBulletinID>
  <sdp:KBArticleID>PMPC-2022-01-25</sdp:KBArticleID>
</sdp:UpdateSpecificData>
```

Trial Catalog Excerpt from cf4a7cef-0f8e-4a1f-99ac-55fad653530b.sdp.

38. Additionally, the software distribution package at the repository component identifies the previous software distribution packages that it replaces, or superseded packages:

```
<sdp:SupersededPackages>
  <sdp:PackageID>cf4b3386-9af0-4f01-a13b-b79723e5abdd</sdp:PackageID>
  <sdp:PackageID>040d4d26-cce8-4a19-a508-701709a8f3ce</sdp:PackageID>
  <sdp:PackageID>e469a22f-07f0-46d0-b54b-7d07ea147972</sdp:PackageID>
  <sdp:PackageID>b1a4ebca-c64a-40d0-932c-bb66fca50ae5</sdp:PackageID>
  <sdp:PackageID>c890fc67-4365-4e15-8082-e2fbb52b0dde</sdp:PackageID>
  <sdp:PackageID>3b12b301-395a-401c-9163-419a4486c4c3</sdp:PackageID>
  <sdp:PackageID>9ec77f12-f5eb-49a4-8ddf-b8e4f263f82c</sdp:PackageID>
  <sdp:PackageID>8ecbe740-a995-4272-ac2b-a0f358a4f2b2</sdp:PackageID>
  <sdp:PackageID>e8f8d63a-0cb4-4903-bfbe-f7f2fc9fd37e</sdp:PackageID>
  <sdp:PackageID>dfc308ee-ec9b-40ff-85cb-05c6e7b402f2</sdp:PackageID>
</sdp:SupersededPackages>
```

Trial Catalog Excerpt from cf4a7cef-0f8e-4a1f-99ac-55fad653530b.sdp.

39. SCCM deploys client agents to the target computers that are used to gather information about the target system as well as perform update installation of software distribution packages.

After you install the client on devices in your organization, Configuration Manager provides several ways to monitor and manage it. You can monitor clients to check their status, and Configuration Manager can automatically fix some problems it detects. Use the Configuration Manager console to manage clients for individual devices or device collections.




- [How to monitor clients](#)
- [How to manage clients](#)
- [Configure the content cache](#)
- [Manage clients on the internet](#)
- [Use collections](#)

Co-management enables you to concurrently manage Windows devices by using both Configuration Manager and Microsoft Intune. It lets you cloud-attach your existing investment in Configuration Manager by adding new functionality. When you enable co-management, you can use Intune for additional client management actions. For more information, see [What is co-management?](#).

<https://docs.microsoft.com/en-us/mem/configmgr/core/clients/manage/monitor-and-manage-clients>.

40. The clients (or discovery agents) can gather hardware inventory information about the target computer system.

## Introduction to hardware inventory

Article • 09/17/2021 • 2 minutes to read •  [Is this page helpful?](#)  

*Applies to: Configuration Manager (current branch)*




Use hardware inventory in Configuration Manager to collect information about the hardware configuration of client devices in your organization. To collect hardware inventory, you must select the **Enable hardware inventory on clients** setting in client settings.

After hardware inventory is enabled and the client runs a hardware inventory cycle, the client sends the information to a management point in the client's site. The management point then forwards the inventory information to the Configuration Manager site server, which stores the inventory information in the site database. Hardware inventory runs on clients according to the schedule that you specify in client settings.

<https://docs.microsoft.com/en-us/mem/configmgr/core/clients/manage/inventory/introduction-to-hardware-inventory>.

41. The clients (or discovery agents) can gather software inventory information about the target computer system.

## Introduction to software inventory in Configuration Manager

Article • 09/17/2021 • 2 minutes to read •  [Is this page helpful?](#)  

*Applies to: Configuration Manager (current branch)*

Use software inventory to collect information about files on client devices. Software inventory can also collect files from client devices and store them on the site server. Software inventory is collected when you select the **Enable software inventory on clients** setting in client settings. You can also schedule the operation in client settings.

<https://docs.microsoft.com/en-us/mem/configmgr/core/clients/manage/inventory/introduction-to-software-inventory>. The client agents gather information about the target computer and relay that

information back to the MEM components discussed earlier. Based on this information and the patch fingerprints, it is determined whether an update should be deployed to the target computer.

**CLAIM 2**  
**INFRINGEMENT OF U.S. PATENT NO. 7,823,147 BY PATCH MY PC**

42. Patch My PC has infringed directly and continues to infringe directly, either literally or under the doctrine of equivalents, at least claim 1 of the '147 Patent by its manufacture, sale, offer for sale, and use of any one or more of the patching services offered by Patch My PC including, but not limited to, Enterprise Plus, Enterprise, and Intune only subscriptions. Defendant is therefore liable for infringement of the '687 Patent pursuant to 35 U.S.C. § 271.

43. Patch My PC received notice of its infringement of the '147 Patent by letter on November 2, 2021. As of the time Patch My PC first had notice of Ivanti's allegations of infringement of one or more claims of the '147 Patent by Defendant, which is no later than November 2, 2021, Defendant indirectly infringed and continues to indirectly infringe at least claim 1 of the '147 Patent by active inducement under 35 U.S.C. § 271(b). Defendant has induced, caused, urged, encouraged, aided and abetted its direct and indirect customers to make, use, sell, offer for sale and/or import one or more of the Accused Instrumentalities, and thus indirectly infringes at least claim 1 of the '147 Patent. Defendant has done so by acts including but not limited to (1) selling such products including features that—when used or resold—infringe, either literally or under the doctrine of equivalents, the '147 Patent; (2) marketing the infringing capabilities of such products; and (3) providing instructions, technical support, and other support and encouragement for the use of such products, including at least the documents referenced above. Such conduct by Patch My PC was intended to and actually did result in direct infringement by its

direct and indirect customers, including the making, using, selling, offering for sale and/or importation of the Accused Instrumentalities in the United States.

44. As a direct and proximate result of Patch My PC's infringement of the '147 Patent, Ivanti has been, is being, and will be damaged. Patch My PC's continued infringement of Ivanti's exclusive rights under the '147 Patent will continue to damage Ivanti, causing irreparable harm for which this is no adequate remedy at law, unless enjoined by this Court.

45. Ivanti is also entitled to recover from Patch My PC the damages sustained by Ivanti as a result of Patch My PC's wrongful acts in an amount subject to proof at trial.

46. As of the time Patch My PC first had notice of the '147 Patent, at least as early as November 2, 2021, Patch My PC has continued with its infringement despite the objectively high likelihood that its actions constitute infringement and Patch My PC's subjective knowledge of this obvious risk. As Patch My PC has no good faith belief that it does not infringe the '147 Patent, at least Patch My PC's continued infringement of the '147 Patent is willful and deliberate, entitling Ivanti to increased damages under 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

47. For example, the Accused Instrumentalities practice and/or are capable of practicing representative claim 1 of the '147 Patent. The following paragraphs provide details regarding only one example of Patch My PC's infringement, and only as to a single patent claim. Ivanti reserves its right to provide greater detail and scope throughout the discovery process.

48. Claim 1 of the '147 Patent recites:

1. A system comprising:
  - (a) a package computer having a plurality of patch fingerprints;
  - (i) wherein the plurality of patch fingerprints includes at least a first patch

fingerprint and a second patch fingerprint, different than the first patch fingerprint;

- (i) wherein at least the first and second patch fingerprints each comprises at least one Extensible Markup Language (XML) metadata query, wherein the first patch fingerprint includes a first XML metadata query, and wherein the second patch fingerprint includes a second XML metadata query, different than the first XML metadata query;
- (ii) wherein at least the first and second patch fingerprints are both associated with a specific software update;
- (b) an update server in communication with the package computer;
  - (i) wherein the update server stores at least the first and second patch fingerprints of the package computer;
  - (ii) wherein the update server is located remote from the package computer; and
- (c) a discovery agent configured to separately interact with both the first XML metadata query and the second XML metadata query to produce first target computer information relating to the first target computer;

wherein the system is configured to:

- (A) send the first XML metadata query and the second XML metadata query of the first and second patch fingerprints from the update server to the discovery agent to gather the first target computer information;
  - (I) wherein the first target computer information is related to at least registry information, software presence information, and software version information relative to the first target computer;
  - (II) wherein a first portion of the first target computer information is associated with the first patch fingerprint and the first XML metadata query;
  - (III) wherein a separate second portion of the first target computer information is associated with the second patch fingerprint and the second XML metadata query;
  - (B) determine, at the update server based on the first target computer information, whether the specific software update is both applicable to and absent from the first target computer;
    - (i) wherein the determination step comprises:
      - (1) evaluating the first portion of the first target computer information to determine the applicability of the specific software update to the first target computer; and
      - (2) if the specific software update is applicable to the first target computer, then evaluating the second portion of the first target computer information to determine the presence or absence of:
        - (A) the applicable files;
        - (B) the applicable registry keys; and
        - (C) the applicable configuration information of the specific software update;
- wherein the system is configured to, based on the determination (B), download the specific software update to one of (i) the update server and (ii) the first target computer.

'147 Patent at 31:31-32:31.

49. The infringing operation of the Accused Instrumentalities detailed above regarding the '660 Patent is also applicable to infringement of the '147 Patent.

50. Further, the Accused Instrumentalities use patch fingerprints, as detailed above, to identify software updates for deployment to target computers. The patch fingerprints include XML metadata queries which are used to determine whether a particular software update is present on a computer and, if not, conditions for deployment to that computer. Additionally, the patch fingerprints include XML metadata queries which are used to determine whether a particular software update is applicable to the target computer.

**CLAIM 3**  
**INFRINGEMENT OF U.S. PATENT NO. 8,407,687 BY PATCH MY PC**

51. Patch My PC has infringed directly and continues to infringe directly, either literally or under the doctrine of equivalents, at least claim 1 of the '687 Patent by its manufacture, sale, offer for sale, and use of any one or more of the patching services offered by Patch My PC including, but not limited to, Enterprise Plus, Enterprise, and Intune only subscriptions. Defendant is therefore liable for infringement of the '687 Patent pursuant to 35 U.S.C. § 271.

52. Patch My PC received notice of its infringement of the '687 Patent by letter on November 2, 2021. As of the time Patch My PC first had notice of Ivanti's allegations of infringement of one or more claims of the '687 Patent by Defendant, which is no later than November 2, 2021, Defendant indirectly infringed and continues to indirectly infringe at least claim 1 of the '687 Patent by active inducement under 35 U.S.C. § 271(b). Defendant has induced, caused, urged, encouraged, aided and abetted its direct and indirect customers to make, use, sell,



offer for sale and/or import one or more of the Accused Instrumentalities, and thus indirectly infringes at least claim 1 of the '687 Patent. Defendant has done so by acts including but not limited to (1) selling such products including features that—when used or resold—infringe, either literally or under the doctrine of equivalents, the '687 Patent; (2) marketing the infringing capabilities of such products; and (3) providing instructions, technical support, and other support and encouragement for the use of such products, including at least the documents referenced above. Such conduct by Patch My PC was intended to and actually did result in direct infringement by its direct and indirect customers, including the making, using, selling, offering for sale and/or importation of the Accused Instrumentalities in the United States.

53. As a direct and proximate result of Patch My PC's infringement of the '687 Patent, Ivanti has been, is being, and will be damaged. Patch My PC's continued infringement of Ivanti's exclusive rights under the '687 Patent will continue to damage Ivanti, causing irreparable harm for which this is no adequate remedy at law, unless enjoined by this Court.

54. Ivanti is also entitled to recover from Patch My PC the damages sustained by Ivanti as a result of Patch My PC's wrongful acts in an amount subject to proof at trial.

55. As of the time Patch My PC first had notice of the '687 Patent, at least as early as November 2, 2021, Patch My PC has continued with its infringement despite the objectively high likelihood that its actions constitute infringement and Patch My PC's subjective knowledge of this obvious risk. As Patch My PC has no good faith belief that it does not infringe the '687 Patent, at least Patch My PC's continued infringement of the '687 Patent is willful and deliberate, entitling Ivanti to increased damages under 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

56. For example, the Accused Instrumentalities practice and/or are capable of practicing representative claim 1 of the '687 Patent. The following paragraphs provide details regarding only one example of Patch My PC's infringement, and only as to a single patent claim. Ivanti reserves its right to provide greater detail and scope throughout the discovery process.

57. Claim 1 of the '687 Patent recites:

1. A method comprising:

- (a) storing at least one patch fingerprint at a package computer; wherein each patch fingerprint comprises at least one extensible markup language (XML) metadata query; wherein at least one of the patch fingerprints is associated with a specific software update;
- (b) downloading the at least one patch fingerprint from the package computer to a repository component of an update server; wherein the package computer is apart from the update server;
- (c) sending the at least one XML metadata query from the update server to a first target computer;
- (d) scanning the first target computer via a discovery agent located on the first target computer, wherein the scanning comprises utilizing the at least one XML metadata query in combination with the discovery agent to produce target computer information; wherein the target computer information is related to at least hardware configuration information, registry information, software presence information, and software version information relative to the first target computer; wherein the first target computer is separated from the package computer via a firewall;
- (e) sending the target computer information to the repository component located on the update server;
- (f) storing the target computer information in the repository component located on the update server;
- (g) comparing, at the update server, at least a portion of the target computer information with at least one of the patch fingerprints;
- (h) determining, at the update server, in response to the comparing step (g), whether the specific software update is absent from the first target computer;
- (i) downloading, in response to the determining step (h), the specific software update to the update server; and
- (j) downloading, in response to the determining step (h) or the downloading step (i), the specific software update from the update server to the first target computer.

'687 Patent at 31:59-32:36.

58. The infringing operation of the Accused Instrumentalities detailed above regarding the '660 Patent and the '147 Patent is also applicable to infringement of the '687 Patent.

**JURY DEMAND**

59. Plaintiff hereby demands a trial by jury on all issues.

**PRAYER FOR RELIEF**

WHEREFORE, Ivanti requests entry of judgment in its favor against Patch My PC as follows:

- a. A judgment that Patch My PC has infringed and is infringing one or more claims of the '660 Patent, either literally or under the doctrine of equivalents;
- b. A judgment that Patch My PC has infringed and is infringing one or more claims of the '147 Patent, either literally or under the doctrine of equivalents;
- c. A judgment that Patch My PC has infringed and is infringing one or more claims of the '687 Patent, either literally or under the doctrine of equivalents;
- d. A judgment that Patch My PC's infringement was willful;
- e. An injunction prohibiting Patch My PC and its officers, agents, employees, and those acting in privity with it, from further infringement of the '660 Patent, the '147 Patent, and the '687 Patent;
- f. An award of damages pursuant to 35 U.S.C. § 284 adequate to compensate Ivanti for Patch My PC's infringement of the '660 Patent, '147 Patent, and '687 Patent in an amount according to proof at trial (together with prejudgment and post-judgment interest), but no less than a reasonable royalty;

- g. An award of costs and expenses pursuant to 35 U.S.C. § 284 or as otherwise permitted by law;
- h. Such other and further relief, whether legal, equitable, or otherwise, to which Plaintiff may be entitled or which this Court may order.

Dated: March 15, 2022

Respectfully submitted,  
*/s/ R. Allan Bullwinkel*

R. Allan Bullwinkel  
Michael F. Heim  
HEIM, PAYNE & CHORUSH, LLP  
1111 Bagby St. Ste. 2100  
Houston, Texas 77002  
Telephone: (713) 221-2000  
Facsimile: (713) 221-2021  
Email: abullwinkel@hpcllp.com  
mheim@hpcllp.com

Emily L. Wasserman  
DAVIS GRAHAM & STUBBS LLP  
1550 17th Street, Suite 500  
Denver, Colorado 80202  
Telephone: (303) 892-7339  
Email: emily.wasserman@dgsllaw.com

**ATTORNEYS FOR IVANTI, INC.**