



Parity 6.0.2 Release Notes

Parity Build: 6.0.2.449
Patch Number: 17
Document Version: 2.18
March 20, 2013

Bit9, Inc.
266 Second Ave, Waltham, MA 02451 USA
Tel: 617.393.7400 Fax: 617.393.7499
E-mail: support@bit9.com
Web: <http://www.bit9.com>

Copyright © 2004-2013 Bit9, Inc. All rights reserved. This product may be covered under one or more patents pending. Bit9 and Parity are trademarks of Bit9, Inc. in the United States and other countries. Any other trademarks and product names used herein may be the trademarks of their respective owners.

Introduction

The Parity 6.0.2 “Release Notes” document provides information for users upgrading from previous versions as well as users new to Parity. It consists of the following major sections:

- **[Before You Begin](#)**: This section describes preparations you should make before beginning the installation process for Parity Server.
- **[Parity 6.0.2: New and Modified Features](#)**: This section describes major changes since 6.0 and should be read by all users.
- **[Corrective Content](#)**: This section describes issues resolved by this release as well as more general improvements in performance or behavior.
- **[Known Issues and Limitations](#)**: This section describes known issues or anomalies in Parity 6.0.2 that you should be aware of.
- **[Contacting Bit9 Support](#)**: This section describes ways to contact Bit9 Technical Support and the information to have prepared to troubleshoot a problem.

This document is a supplement to the main Parity documentation.

Important Information

Versions of Parity at Patch 17 and above contain important changes to the handling of Bit9’s digital certificates. We strongly recommend that you upgrade the Parity Server and then upgrade all Parity Agents from prior versions as soon as is practical. Please contact Bit9 Support if you have additional questions.

About Your Shipment

Your Parity distribution includes the Parity Server installation program and documentation files. Parity Server custom-generates custom agent installation packages at your site for each protection policy you define, so no separate agent installer is needed in the original distribution.

Documentation

Your Parity documentation set consists of online PDF and HTML files included with the product distribution and also available in the [Bit9 web site](#) Support area.

- **Installing Parity** -- Provides instructions for installing and configuring the Parity Server and Parity Agent.
- **Using Parity** -- Describes Parity operation, including step-by-step instructions for administration and configuration tasks. Management topics for computer systems, including agent installation, are also covered.
- **Parity Events: Integration Guide** – Describes the events that are generated, tracked, stored, and accessible through the Parity system, and the ways you can access Parity event data outside of the Parity Console user interface.

[Back to Table of Contents](#)

Before You Begin

This section describes preparations you should make before beginning the installation process for Parity Server. These include actions you should take before installing Parity Server, preparations you should make for configuring the server after installation, and general information you should know about server and agent. It contains information that applies to upgrades and new installations.

System Requirements

The most current *Operating Environment Requirements* for Parity 6.0.2 are provided in a separate document which is readily available from Bit9 Support. Hardware and software requirements also are documented in the *Installing Parity* guide provided with Parity 6.0.2. Both upgrade and new customers should be sure to meet the requirements before beginning.

Platform Downloads

This section contains links to download additional software that may be required to install Parity version 6.0.2. Consult the *Installing Parity* guide for more information.

Windows Installer 4.5:

<http://www.microsoft.com/downloads/details.aspx?FamilyID=5a58b56f-60b6-4412-95b9-54d056d6f9f4>

SQL Server 2008 Express:

<http://www.microsoft.com/express/Downloads/Download-SQLE.aspx>

Parity Server Upgrades

This section is for upgrades only. If you are not upgrading, see [New Parity Installations](#) (page 6).

Support for the Upgrade Process

Parity Server and Agent upgrade support is covered under the Customer Parity Maintenance Agreement. Bit9 recommends contacting Technical Support prior to performing the upgrade for further details on the upgrade process and the latest information that supplements the information contained in this document. Technical Support is available to assist with the upgrade process to ensure a smooth and efficient upgrade installation.

Before Running the Server Upgrade

The following tasks should be done *before* you run the Parity Server upgrade program. Note that some tasks only apply to upgrades from Parity 5.x.

- **(Upgrades from 5.x) Replace Parity-Installed SQL Server:** If you are upgrading from Parity 5.x and currently are using a version of SQL Server Express that was installed by the Parity installer, you will need to provision SQL Server separately using your own licensed copy. Please contact Bit9 Support for more information.
- **(Upgrades from 5.x) Pure Mode Required:** Before you can upgrade to Parity 6.0, you must move your Parity Server into Pure Mode. Consult the Parity 5.0 documentation or online Help for more information.
- **Backup Parity Server Database:** Backup your Parity Server database before you begin the upgrade process. Backup is disabled during upgrade and must be re-enabled once you are sure the upgrade was successful.
- **Backup Certificates Separately:** Parity Server's backup mechanism does not backup the certificates associated with Parity. Please back those up separately on a system other than the Parity Server.
- **Disable Distribution Systems:** If you use third party deployment mechanisms (e.g. SCCM), disable the distribution of the Parity Agent before you begin the Agent upgrade process. This will prevent attempts to "downgrade" the Parity Agent.
- **(Upgrades from 6.0.0) Review Trusted Directory Volume:** When you upgrade from Parity 6.0.0 to 6.0.2, Trusted Directories are reprocessed for all agents as soon as the Parity Server is upgraded. If you have a large amount of content in Trusted Directories, consider scheduling the Parity Server upgrade for a time when this will not be disruptive. Upgrades from v5.x reprocess the directories for agents only when each *agent* upgrades.

Prepare for Post-Upgrade Tasks

You should be prepared to do the following tasks after you run the Parity Server upgrade program.

- **Review Directory Policies/Custom Rules:** “Directory Policies” in Parity 5.x are converted to “Custom Rules” in Parity 6.0 and 6.0.1. After you have upgraded Parity Server, review the Custom Rules tab on the Software Rules page. The Parity Server upgrade process migrates existing Directory Policies to this new format. Make sure the rules are configured as you want them to be.
- **Review Syslog Format:** New Syslog formats have been added in Parity 6.0.1 and 6.0.2 and event formats were improved in 6.0.1. If you use Syslog, you should review the available choices and ensure that any additional processing of Syslog messages can support the improved events.
- **Review External Event Settings:** If you use External Events, review the settings to ensure they are still enabled and correctly functioning.
- **Review Updaters:** New Updaters have been added in Parity 6.0.2. Review the Updaters tab on the Software Rules page to make sure the correct updaters are enabled.
- **Update Agent Distribution Points:** If you use third party deployment mechanisms (e.g. SCCM), re-enable or re-create them using new agent packages from the upgraded Parity Server.
 - Use **Parity6.0MaintenancePack.msp** to upgrade from Parity Agent 6.0 or 6.0.1.
 - Use **ParityHostAgent.msi** to upgrade from a pre-6.0 agent.

- **Review the New Parity Installations section:** Although it is for new installations, this section also includes information of possible interest to upgrade customers.

New Parity Installations

This section describes preparatory tasks and suggested post-installation tasks for new Parity Server installations. Although targeted at new installations, it should be reviewed by new and upgrade customers.

Prepare for Parity Server Installation

- **Choose Account for Parity Server Installation:** Bit9 recommends that you use a Domain Service Account for Parity Server installation. If you plan to use Active Directory services or use an authenticated proxy to access the Internet, a Service Account is required. This account must be assigned Local Administrator privileges on the Parity Server.
Note: Do not change the permissions level of the account with which you install Parity after installation.
- **Review .NET Configuration:** If Microsoft .NET 4 is installed on your Parity Server system with Windows 2008 Server, ensure that the IIS DefaultAppPool is set to use “.NET Framework v2.0.50727” by default.
- **Prepare to Enable Parity Agent Management Access:** The Parity Agent Management screen in the new installation dialog allows you to designate a user or group, or a password usable by anyone, to perform certain agent management activities assisted by Bit9 Technical Support. Especially if you will have client computers that will never be connected to Parity Server, it is best to set up a client access option before generating and distributing agent installation packages. If you are unable to configure access during installation, you can do it later on the Management Configuration page in Parity Console. See the *Using Parity* manual (or online help) for more details.

Prepare for Post-installation Tasks

- **Enable Parity CLI Management Access:** If you did not enable Parity Agent Management access during installation, go to the Management Configuration page in Parity Console to enable it, preferably before deploying agents. See “Management Configuration” in the *Using Parity* manual (or online help) for more details.
- **Confirm Agent Installation Privileges:** The Parity Agent installer must be run either by Local System or a user account that has administrative rights and a loadable user profile.
- **Install Agent as Per-Machine Application:** The Parity Agent should only be deployed as a per-machine application, not per-user.
- **Consider Agent Rollout Impact:** As soon as the Parity Agent is installed, it connects with the server and begins initializing files. Because initialization can involve an increased flow of data between the Parity Server and its new client, be sure your agent rollout plans take your network capacity and number of files into account — simultaneous agent installation on all the computers on a large network is not recommended.
- **Review Trusted Updaters:** Review Trusted Updaters to ensure the correct ones are enabled for your environment before you begin large-scale Parity Agent deployment.

- **Review Root Certificates for Trusted Publishers:** Trusted Publishers are validated by Windows. For properly validation to occur, the correct, up-to-date root certificates must be installed for these publishers. You should ensure that Microsoft root certificate updates are included in your Windows Updates. If you plan to use in-house certificates, ensure that your in-house root certificates are installed on each endpoint on which you will install Parity Agent.
- **Consider Allowing Expired Certificates for Publisher Validation:** By default, Parity does not allow approval-by-publisher of software with expired certificates. This provides extra security, but can prevent approval of legitimate files whose valid certificate is now out of date. Parity offers the option of trusting expired certificates whose (verifiable) timestamp is within the certificate validity period. If possible, make this decision before installing agents that will seldom or never be connected to the Parity Server. See “Approving by Publisher” in *Using Parity* (or online Help) for more information.
- **Test User-supplied Certificates:** Parity Server allows you to enable the use of user-supplied certificates for Parity Agent-Server communication. This is a one way process and cannot be reversed. To validate its connection to the Parity Server, each agent system must have up-to-date root certificates. Bit9 recommends that you test your new certificates before large-scale Parity Agent deployment begins. See “Securing Agent-Server Communications” in the *Using Parity* manual or online Help for more details.
- **Prepare Notifier Text and Link Changes:** If you intend to change Notifier text or links for most or all Parity policies, make your changes in the Template Policy *before* you create other policies. This will save time and effort.
- **Specify a Custom Notifier Logo if necessary:** You can specify a custom logo for the notifier that appears when Parity blocks an action on an agent computer. This is done on a per-policy basis. See “Specifying a Custom Notifier Logo” in *Using Parity* (or online Help).
- **Review Content of Trusted Directories for Distribution Systems:** If you use Windows Software Update Services (WSUS) or other software distribution mechanisms (e.g. SCCM or Altiris), pre-approving this content with a Trusted Directory before large-scale Parity Agent deployment will ensure a more effective transition to Lockdown.
- **Java Tracking:** Support for tracking Java class and jar files is not enabled by default. If you plan to track Java applications, please contact Bit9 Support for more information.
- **Exclude Parity Agent from AV scanning:** Antivirus products should be configured to exclude the following from on-access scanning:
 - the Parity process (**Parity.exe**)
 - the Parity program directory (**Program Files\Bit9** on 32-bit systems and **Program Files (x86)\Bit9** on 64-bit systems)
 - the Parity data directory (**ProgramData\Bit9\Parity Agent** on Vista, Windows 7 and Windows 2008 systems and **\Documents and Settings\All Users\Application Data\Bit9\Parity Agent** on other supported systems)
- **Consider Other Agent Interactions:** Certain other types of software may interact with Parity Agent – contact Bit9 Support for more information on each of these cases:
 - Disk encryption software may interact with the Parity Agent. In general, full disk or partition encryption should minimize the chances of problems. However, some encryption products are compatible with Parity with other types of encryption (file or folder) enabled.
 - Ghosting or imaging systems with Parity pre-installed requires additional steps on the master system. This is because the Parity Agent uses a unique identifier when communicating with the Parity Server. For more information, go to the Bit9 Support web site and see Solution 207 (“How to Create an Image with Parity Installed”).

- **Do not Change SQL Recovery Model:** Parity sets the Simple Recovery Model for the Parity SQL Database. Do not change this.

[Back to Table of Contents](#)

Parity 6.0.2: New and Modified Features

The following section provides a quick reference to the feature changes made between 6.0.1 and 6.0.2. If you are upgrading from pre-6.0 versions of the product, you might also be interested in the separate document *Introduction to Parity v6.0*, which details changes made for that release, and the document *Parity 6.0.1 Release Notes*, which details changes made for that release.

New Memory Protection Rules

Parity 6.0.2 includes two new Memory Protection rule types:

- **Dynamic Code Execution:** This protection prevents arbitrary or floating code execution of the sort used by many forms of malware and protects against attempts to disable Dynamic Execution Protection (DEP). Applies only to 32-bit systems.
- **Kernel Memory Access:** This protection prevents applications from accessing kernel memory but allows you to grant exceptions for legitimate, approved applications. Applies only to Windows XP and Windows Server 2003 (without SP1).

See the "Memory Rules" chapter in *Using Parity* (or online Help) for more information.

Improved Agent Hardening

Parity 6.0.2 includes a number of enhancements that make it more difficult to circumvent or defeat Parity protection on agent computers.

Updater Changes

This release includes the following changes to the Updaters you can use for automatic approval of application and software distribution updates:

- **Windows Update enabled by default:** The updater for Windows Update is enabled by default for 6.0.2 Parity Agents and does not appear as an option for new Parity Server 6.0.2 installations. It is listed for Parity installations with agents older than 6.0.2, and it must be enabled to allow Windows Update to run for those older agents.
- **New Updaters:** The following new updaters are available in 6.0.2:
 - Google Apps
 - Google Chrome
 - WebEx for Chrome
 - Microsoft Security Essentials
 - Java and Bundled Software (allows updates that install or update add-ons, such as search bars or third-party applications, included in some versions of Java)
 - Updaters have been added for newer versions of certain products.

See the "Approving by Updater" section in *Using Parity* (or online Help) for more information.

Computers Page Enhancements

The Computers page in the Parity Console has the following new features:

- **Current Configuration List:** The current CL (configuration list) version number now appears on the Computers page. This allows you to easily compare the latest version on Parity Server with the version on each agent.
- **Unprotected computer indicator:** A new red dot indicator appears next to the name of a computer if it is not protected because of problems loading drivers. This signals that the agent computer should be rebooted to allow the drivers to load.

See the "Managing Computers" chapter in *Using Parity* (or online Help) for more information.

Notifier Enhancements

Parity 6.0.2 includes two enhancements to the blocked file notifier on agent computers:

- **Custom notifier logo:** In previous releases, you could choose to display the Bit9 logo or no logo on the notifier that appears when a Parity rule blocks an action. You can now specify a custom logo on the notifier instead of the Bit9 logo or no logo. The logo option is specified on a per policy basis.
Note: If you set up a custom notifier logo using a special implementation in a previous release, you must update that to the new, supported implementation in 6.0.2.
- **Promote file option:** There is a new, hidden feature on block-and-ask notifiers on agent computers. If the user presses the **Shift** key, the "Allow" button changes to a "Promote" button. Promote ensures that the file runs as a promoted process, meaning that files written by the process will be locally approved. This is useful if the notifier is displayed for an execution attempt on a file that initiates an installation but might not be recognized by Parity as an installer.

See the "Creating and Configuring Policies" chapter in *Using Parity* (or online Help) for more information about notifier options. See the "Getting File Information" chapter for more about installer files and promotion.

Live Inventory SDK Modifications

The following changes have been made to the Live Inventory SDK for this release:

- In the ExComputers view, the **User_Id** column was removed since it was redundant with Users column.
- In the ExComputers view, the **Computer_Deleted** column was added.

Beginning with build 6.0.1.162, the following elements also became available:

- In the ExFileCatalog view, **First_Seen_Computer_id** is a foreign key into the ExComputers table for the computer on which the file was first seen.
- In the ExFileInstances view, **Group_Type** specifies which of the available grouping mechanisms was used to assign this file instance to a group.

See "Live Inventory SDK: Database Views" in *Using Parity* (or online Help) for detailed information about the public views into Parity's live inventory database.

New Agent Management Screen during Installation

In previous releases of Parity, you could enable global access to the command line interface for agent management through the System Configuration page on Parity Console. Agent management commands can be used for diagnostics, recovery, and other special situations.

For new Parity 6.0.2 installations, you can configure agent management command access during installation, on the Parity Agent Management screen. This provides a convenient way to configure access via a global password or a specified user or group, *before Parity creates agent installation packages* so that your access choice is built into the agents. It is especially important to set up a global access method if you will have agents that are offline frequently or at all times.

See the *Installing Parity* guide or the “Configuring Agent Management Privileges” section in *Using Parity* (or online Help) for more information about configuring agent management access.

Syslog Enhancement: Support for Q1Labs LEEF Integration

Parity Server now supports Syslog output for Q1Labs LEEF (Log Event Extended Format). Using Q1Labs LEEF, you can integrate Parity event logs with QRadar SIEM or QRadar Log Manager.

See “Mapping Parity Events to Q1Labs LEEF Format” in *Parity Events: Integration Guide* for more information about QRadar integration.

In the Parity Console, you configure Syslog integration on the System Configuration/Server Status page, described in the “Parity Configuration” chapter of *Using Parity* (or online Help).

Parity Events: Integration Guide

The *Parity Events: Integration Guide* is now part of the standard suite of documentation available with Parity. This document describes all of the events generated by Parity, and the different ways you can use event output. It also highlights any event-related changes in this release. The document is available as part of your shipment or download of Parity 6.0.2. It is also available on the Parity Support web site.

Because the events guide is now standard, detailed information about specific Syslog integrations supported by Parity will appear there instead of in *Using Parity*. This includes **ArcSight (CEF)** integration and **Q1Labs (LEEF)** integration.

[Back to Table of Contents](#)

Corrective Content

Each release includes general improvements in product quality, based on our on-going testing of Parity 6.0.2.

Corrective Content in Parity 6.0.2 GA Release (Build 448, Patch 17)

- Improvements in Offline Certificate Validation
 - Details: In previous releases, when certificates could not be successfully validated, for example on permanently offline systems, a fallback approach checked an internal revocation list. In some circumstances, this list was applied incorrectly. In this release, the certificates on this internal list are correctly handled. [Agent]
- Enhancements in Digital Certificate Processing
 - Details: The digital certificate used to sign prior releases has been revoked. In addition to using a newly issued certificate, this release will explicitly unapprove any software that was previously signed by the revoked certificate, even when “Bit9” is a Trusted Publisher. This prevents any software signed by this certificate from running in Lockdown (SecCon 20) or Block and Ask (SecCon 30). [Agent, Server]
- Enhancements in Trusted Directory Robustness
 - Details: In rare circumstances, processing of items in a Trusted Directory would fail when these items contained malformed data. This release improves the handling of such situations. [Agent]

Corrective Content in Parity 6.0.2 GA Release (Build 440, Patch 16)

- Performance Issues with Network Files
 - Details: When accessing files over a network, the Parity agent would cause a marked slowdown in the performance of certain operations, such as copying files. By caching additional internal information, this release improves the performance of operations on network files. [Agent]
- Improvements in Console Security
 - Details: Several issues affecting the security of the Parity console were addressed in this release. [Server]

Corrective Content in Parity 6.0.2 GA Release (Build 427, Patch 15)

- Crash Processing Downloaded *configlist* File
 - Details: Certain errors in downloading a *configlist* file from the Parity Server would cause the Parity agent to crash, sometimes resulting in repeated downloads of the file. This release correctly handles the download errors so that they do not result in a crash. [Agent]

Corrective Content in Parity 6.0.2 GA Release (Build 424, Patch 14)

- Deadlock on High Volume Servers
 - Details: In previous releases, a Windows kernel deadlock would occur when Parity was run on high-volume servers. In this this release, the identified deadlock condition has been eliminated. [Agent]
- Parity Reporter Stops Processing Tasks
 - Details: An issue in the Parity Reporter occasionally caused it to stop processing periodic tasks. These include the processing of “Files on Computers” information, backups and other low priority tasks. This release fixes the underlying issue that was preventing the Parity Reporter from correctly scheduling and processing these tasks. [Server]
- Blocks Reported on *upx.exe* and *crawler.exe*
 - Details: The Parity Agent would report tamper protection blocks on *upx.exe* and *crawler.exe*, even though these are Parity executables. This release does not report these blocks. [Agent]

- **Inconsistent Information for Parity Components in “Files on Computers”**
 - Details: During an upgrade, Parity components are modified, but these changes are not tracked by Parity. This can lead to inconsistent information for these components in “Files on Computers.” This release ensures that up-to-date information is sent to the Parity Server for upgraded Parity Agent components. [Agent]
- **Macros are not Auto-completed in Custom or Memory Rules**
 - Details: In the Parity Console, typing the “<” character when entering Custom or Memory Rules should auto-complete with available Parity macros. In some circumstances, no auto-completion would occur. This release auto-completes macros in all appropriate Parity Console fields. [Server]
- **Filter Changes to Saved Views are Lost**
 - Details: When filters or groupings were added to a Saved View, these changes would be lost when navigating away from the page and later returning to the same page. This release corrects this issue. [Server]
- **Resetting Filters does not Reset Saved Views**
 - Details: “Reset Current Settings” in a Parity Console user’s preferences did not reset the Saved View. In this release, saved filter settings are cleared. [Server]
- **“&” replaced with “&” on Computer Details**
 - Details: If “&” was used in the Computer Tag or Description fields on the Computer Details page, it would be replaced with “&” when the details were saved. The “&” character is now correctly preserved in these fields. [Server]
- **Unicode Characters Displayed Incorrectly for Active Directory Users**
 - Details: When an Active Directory user logged into the Parity Console, certain Unicode characters in the user’s name were displayed incorrectly. In this release, the characters are now displayed as they appear in the Active Directory. [Server]
- **Unicode Characters not Supported by Active Directory Policy Mappings**
 - Details: When browsing the Active Directory to create Policy Mappings, certain Unicode characters could not be selected in the Active Directory browser. This release allows mappings to be created from Active Directory objects that contain any Unicode character. [Server]

Corrective Content in Parity 6.0.2 GA Release (Build 412, Patch 13)

- **Unexpected Processes Trusted**
 - Details: In some circumstances, processes would become trusted when this was not intended. This release eliminates additional cases of unintended trust and ensures they will not happen in the future.
- **Notifier Crashes when Link Selected**
 - Details: Due to interactions with Parity tamper protection, clicking on the link in the Notifier dialog would cause the Notifier to crash in some circumstances. In this release, tamper protection does not interfere with the Notifier link.
- **Console Passwords Cannot Contain ‘&’**
 - Details: Parity Console passwords containing the ‘&’ character would not function correctly. This included passwords from Active Directory accounts. In this release, passwords containing the ‘&’ character are accepted by the Parity Console and function correctly.
- **Computers Hang on Reboot after Parity Agent Installation**
 - Details: In very rare circumstances, computers on which the Parity agent had recently been installed would hang the first time they were rebooted. This release corrects an issue with the agent that caused a deadlock at boot time.
- **Parity Server Crash Due to Network Error**
 - Details: Occasionally, the Parity Server would crash and restart when it received malformed data from Parity agents. This occurred when there was an underlying network error. In this release, the Parity Server logs the error and continues operation.
- **Backup Issue After Moving from Local to Remote Database**
 - Details: When following the standard procedure for moving databases to a remote SQL Server, the Parity Server installer would incorrectly flag the database as local, causing Parity backups to fail. This release flags the database as remote, allowing backups to proceed correctly.

- Parity Driver Information Cleanup
 - Details: Occasionally, the Parity upgrade process would leave incorrect information in the Windows Driver Store. This release ensures that the Driver Store contains only the correct information, allowing future upgrades to proceed more smoothly.

Corrective Content in Parity 6.0.2 GA Release (Build 403, Patch 12)

- Unexpected Processes Trusted
 - Details: In some circumstances, processes would become trusted when this was not intended. This release eliminates reported cases of unintended trust.
- Certain Components not Approved in Offline Upgrade
 - Details: When the Parity Agent is upgraded, it may also upgrade its internal database. Under some circumstances, this prevented approval of certain files when upgrading offline endpoints. This release addresses this issue.
- Stop Errors when File Activity High
 - Details: On some systems with a high volume of file activity, such as Exchange Servers, a stop error occurred in rare circumstances. This release corrects that error.
- Information not Synchronizing with Parity Knowledge
 - Details: An issue that was preventing the Parity Server from synchronizing file information with Parity Knowledge was fixed in this release.
- 100 Character Limit on Notifier Link
 - Details: In previous releases, the Notifier link was limited to 100 characters. In this release, you can specify a link of up to 2047 characters.
- Notifier Links Containing “<” or “>” Not Saved
 - Details: Links containing the characters “<” or “>” could not be saved. This release allows such links to be saved.

Corrective Content in Parity 6.0.2 GA Release (Build 393, Patch 11)

- Deleted Users Appear in Exported CSV
 - Details: When a list of users was exported from the “Login Accounts: Users” screen, deleted users would appear in the list. In this release, deleted users are no longer included in the exported CSV file.
- Parity Agent Crashes when Computer Cloned
 - Details: In some circumstances, cloning a computer that was running a Parity Agent would cause the agent to crash. Computers running Parity Agents can be cloned successfully in this release.
- Kernel File Operations Incorrectly Processed
 - Details: File operations that originated in the Windows kernel were not being correctly processed by the Parity Agent, which would affect the ability to use file control rules. This release correctly processes these kernel operations.
- Wildcard Expansion of Symbolic Paths
 - Details: Because wildcards in symbolic paths were not being correctly expanded, certain approvals that should have occurred because of Windows Update and other Trusted Updaters were not occurring. This caused blocks on components in the SxS folder or in components installed by software management systems. This release handles the wildcard expansion correctly.
- Error Banning a File with Unicode Name
 - Details: Files with long Unicode names could not be banned in the Parity Console. This release allows these files to be banned.
- Cannot Open “File Details” Page with a Right Click
 - Details: When a list of files was displayed in the Parity Console, it was not possible to right click on the associated “details” link to open the detailed information in a separate browser tab or window. This release allows the Parity Console user to use the right click.
- Parity Console Times Out Editing Custom Rule
 - Details: The Parity Console would time out while editing certain Custom Rules. This release allows all Custom Rules to be correctly edited without a timeout.

- Parity Console Database Timeouts
 - Details: An internal maintenance procedure would occasionally cause database timeouts. This procedure has been redesigned in this release to avoid any adverse effect on the Parity Console.
- Executable Files Incorrectly Marked as an Installer
 - Details: Certain executable files that contain a single archive were incorrectly marked as installers. In this release, this issue is corrected. Note that an upgrade to this release does not retroactively correct existing executable files. Each of these must be marked as “Not an installer” in the Parity Console to correct the issue.
- Parity Server Patch Utility Did Not Update Parity Console
 - Details: The Parity Patch Utility did not install all required files for some previous Parity Console corrective content. This release installs all prior corrective content for the Parity Console.

Corrective Content in Parity 6.0.2 GA Release (Build 384, Patch 10)

- Parity Agents not Connecting after Server Resends Policy Rules
 - Details: Occasionally, if the *Resend Policy Rules* option had been used for a Parity Agent, the agent would no longer connect to the Parity Server. This was caused by an issue that did not allow the Parity Agent to register correctly with the Server. In this release, the agent correctly registers and the resending of policy rules is completed successfully.
- Stop Errors Caused by Paged Pool Exhaustion
 - Details: When there a large number of processes were running, Parity could cause the system to exhaust the paged pool, leading to a Stop Error. This release corrects an additional cause of this problem that was identified after the previous release.
- Missing Event for Tracking Temporary Override Codes
 - Details: When a Temporary Override Code was generated, no event was generated in the Parity Console. In this release, an event is generated that includes the target SecCon and the user who created the override code.
- Parity Time Zone can be UTC
 - Details: It is now possible to use Coordinated Universal Time (UTC) for the time zone in the Parity Console. This setting is available in the *Server Status* settings on the *System Configuration* page.

Corrective Content in Parity 6.0.2 GA Release (Build 378, Patch 9)

- Block and Ask Notifier Not Displaying, Causing Hang
 - Details: When a macro expansion of Notifier text resulted in an empty message, the Notifier would not display. In Block and Ask mode, this would cause the underlying executable to hang, since no response could be given. In this release, the Notifier displays even if the text is empty.
- Cloned Systems Run Unnecessary Cache Consistency Check
 - Details: When a system was cloned, the clone would run a cache consistency check at boot time, whether this was necessary or not. This release will not run a consistency check after the newly cloned system boots.
- Stop Errors Caused by Paged Pool Exhaustion
 - Details: In circumstances where there were a large number of processes running, Parity could cause the system to exhaust the paged pool, leading to a Stop Error. This release corrects one cause of this problem.
- Kernel Stack Overflow Causes Stop Error
 - Details: In some circumstances, Parity would fail to detect a likely kernel stack overflow and not correctly avoid this situation. Since kernel stack is a limited resource, Parity detects potential overflows. This release improves the detection for stack overflows.
- SQL Exception in *DailyPruneTask*
 - Details: In some circumstances, Parity Server attempted to run *DailyPruneTask* without having the appropriate permissions, causing a SQL exception. Permissions are now checked in this release, and the statistics associated with this task are not gathered if proper permission is not available.

- **Unexpected Blocks on *rundll32.exe***
 - Details: Block events were reported for *rundll32.exe*, which appeared to be attempting to run certain Parity components. These were false positives, not actually execution blocks. In this release, these blocks are no longer reported.
- **Data Execution Prevention (DEP) Causes Parity Components to Crash**
 - Details: When other processes attempted to inject DLLs incompatible with Data Execution Prevention (DEP) into Parity or its components, these components would crash. Parity enables DEP in all its components, so that injection of any DLL that is not DEP compatible causes an issue. In this release, injection of DLLs that are known to cause problems is silently prevented.
- **Block Events Not Being Reported**
 - Details: In circumstances where large numbers of events were being reported, some events would not be reported to the Parity Console. This did not affect the operation of the Parity Agent, which continued to enforce policy correctly. This release correctly reports all these events.
- **Blocks with Windows Defender Updates**
 - Details: Certain Windows Defender updates were blocking, even when the Windows Defender Trusted Updater was enabled. This release allows future Windows Defender updates to install and run correctly.
- **File Grouping with the BigFix Updater**
 - Details: In previous releases, the BigFix Updater would group all files into one large group. Starting in this release, the Updater groups files separately for each installation.
- **Live Inventory SDK ExFileInstanceGroups View**
 - Details: In some cases the ExFileInstanceGroups public view was missing entries. This release addresses the issue.

Corrective Content in Parity 6.0.2 GA Release (Build 369, Patch 8)

- **Parity Agent Can Be Disabled by Removing Permissions**
 - Details: Removing permissions for the System user from certain Parity registry keys could be used to disable the Parity agent. This issue is addressed in this release.
- **Parity Agent Can Be Disabled with Hardware Profiles**
 - Details: The Parity Agent could be disabled using the Hardware Profiles feature of the Services Control Panel. This is no longer possible in this release.
- **Software Installation Failures (Microsoft Office, SCCM)**
 - Details: In some circumstances, installation of large packages, such as Microsoft Office, or installations via Microsoft SCCM would fail. This was due to an issue with file sharing violations that has been addressed in this release.
- **Interaction Between Data Execution Protection and DLL Injection**
 - Details: Third-party software that attempted DLL injection into secondary Parity processes, such as the Notifier, would cause these Parity processes to be terminated by Data Execution Prevention (DEP). This release correctly prevents DLL injection for secondary processes.
- **Parity Server Installation Fails with Special Characters in Password**
 - Details: When the Parity Server was installed using an Active Directory account for which the password contained special characters (such as ">"), the installation would fail. This release allows passwords containing special characters.
- **Performance Issue Pruning Non-Prevalent Files**
 - Details: SQL Server performance would degrade when the Parity Server option for pruning non-prevalent files from the Parity Server database was used. This release adds indexes to the database to improve the performance of this pruning operation.
- **Parity Agent Fails to Complete Initialization**
 - Details: When the list of available disk volumes changed dynamically, the Parity Agent would not complete initialization. In this release, dynamic volumes are correctly accounted for.

- **Uninstall-Reconnect of Server Set Incorrect Agent Upgrade File Permissions**
 - Details: When Parity Server was uninstalled and then reinstalled, permissions for Parity Agent upgrade files were not set correctly, which caused agent upgrade issues. Agent upgrade file permissions are set correctly in this release. Note: This issue was seen only when the server was completely uninstalled and then reinstalled; use of the Parity Server Patch mechanism did not interfere with agent upgrades.

Corrective Content in Parity 6.0.2 GA Release (Build 356, Patch 7)

- **Privilege Escalation Allows Execution of Unapproved or Banned Executables**
 - Details: A vulnerability in Parity allowed an attacker to inject malicious code into the Notifier process and potentially run software that would otherwise block. The attack required that any software used in the attack was already approved. This vulnerability is addressed in this release.
- **Excessive Resource Consumption During Cache Validation**
 - Details: In rare cases, Parity Agent cache validation, which must occur before the agent begins operation, would consume large quantities of CPU and never be completed. This prevented the agent from appearing online. In this release, the cache validation completes successfully.
- **Notifier Did Not Appear when Closing Tag is Missing**
 - Details: When a Parity rule blocked an action, no Notifier was displayed if a closing tag (“>”) was missing in the Notifier text. Although any Pending or Banned files were correctly blocked, the Parity Agent also consumed more CPU than normal. In this release, the Notifier is displayed even when the closing tag is missing, displaying the message “[Format Errors]”. Also, the agent does not consume additional CPU in this case.
- **Interaction with IBM BigFix**
 - Details: On certain platforms, installation of the IBM BigFix agent prevented the Parity Notifier from being displayed. This release works around this incompatibility.
- **Incorrect Data in Baseline Drift Reports**
 - Details: When a comparison method of “File content” was chosen for a Baseline Drift report, hashes that appeared in both the Target and the Baseline were incorrectly included in the report of differences in some cases. In this release, these false differences are not included in the report.
- **“Show” Menu on the “Disconnected Computers” Tab**
 - Details: The “Show” menu on the “Disconnected Computers” tab on the “Computers” page did not allow choice of “All” disconnected computers after another time-based setting had been chosen. The menu now allows “All” to be chosen.
- **Parity Server Patch Failure on Large Databases**
 - Details: The Parity Server patch installer sometimes failed on larger SQL Server databases. This was due to occasional timeouts accessing the database. This release increases the timeout.
- **Parity Server Patch Failure with Local Users**
 - Details: If a Local User account was used for the original installation of Parity Server, the patch installer would fail. In this release, the patch can be correctly installed in these cases.
- **Certain Dashboard Portlets Do Not Refresh**
 - Details: Some Dashboard portlets did not refresh unless the entire Dashboard was refreshed. This issue is addressed in this release.
- **Event Documentation Updated to Include Event Subtype Numbers**
 - Details: The “Parity Events: Integration Guide” has been updated to include ID numbers for event subtypes.

Corrective Content in Parity 6.0.2 GA Release (Build 339, Patch 6)

- **Windows Update Approvals Occasionally Fail**
 - Details: In rare circumstances, Parity failed to approve certain files written by Windows Update. This release allows correct approval of files from future Windows Updates. Files created by Windows Update on computers running prior versions of Parity may require manual approval in the Parity Console.

- Parity Blocks some Digitally Signed Executables
 - Details: The certificate signing time for executable files that did not contain counter signatures was not being correctly determined, causing Parity to block the files, even when a Trusted Publisher was approved. After the Parity Agent is upgraded to this release, newly discovered executable files will be correctly processed. For existing files, a “Full scan for new files” using “Re-evaluate publishers” is required to rectify this issue.
- Notifier Link Field Truncated
 - Details: In the Parity Console, a Notifier link can be specified on the Edit Policy page or the Edit Settings page. This link field was incorrectly truncated in previous releases on the Edit Settings page, but is now permitted to be the full supported length.
- Server Installation Fails due to NTFS Check
 - Details: Because of an issue with a Microsoft utility, the Parity Server installer can fail to determine that an NTFS filesystem is present. This caused Parity Server installation to fail. This release provides a mechanism for Bit9 Support to disable this check. If you believe you have encountered this issue, please contact Bit9 Support for assistance.

Corrective Content in Parity 6.0.2 GA Release (Build 324, Patch 5)

- Notifier Does Not Display on Windows Server 2003 64-bit over RDP Connections
 - Details: Due to issues in Windows Server 2003 64-bit, the Parity Notifier would not display when used over a Remote Desktop (RDP) connections. In this release, the Parity Notifier has been modified to work-around this Windows limitation. Please contact Bit9 Technical Support for instructions on how to configure the Notifier for RDP connections.
- Windows Server 2003 64-bit Occasionally Hangs on Agent Upgrade
 - Details: During the Parity Agent upgrade process, a user logon to the computer could cause a deadlock in the system, resulting in a hang. This deadlock is resolved in this release.
- Per-Policy Global Approvals are not in Effect on the Agent
 - Details: In some circumstances, Global Approvals that apply to specific Parity Policies would not be accepted by the endpoints. This issue is addressed in this release.
- Write Blocks on Approved Removable Media
 - Details: When certain removable media were approved in Parity, write blocks for files on the media would continue to occur. Some instances of this problem have been addressed in this release.
- Inconsistent State on Hashes with More than One Instance
 - Details: When a particular approved hash had more than one instance on the file system, there were circumstances in which only one instance would be marked as approved. In this release, all instances are properly approved.
- BigFix Documentation Updated
 - Details: Documentation for BigFix integration with Parity is updated in this release to reflect the acquisition of BigFix by International Business Machines.
- Inconsistencies in Parity Agent Cache could Lead to Incorrect Operation of Parity
 - Details: A problem introduced in Patch 4 could lead to inconsistencies in the Parity Agent cache and incorrect operation of Parity. This issue is addressed in this release.

Corrective Content in Parity 6.0.2 GA Release (Build 310, Patch 4)

- Parity Agent Installation Packages not Digitally Signed
 - Details: If Parity was updated via the Patch process, the Parity Agent installation packages (*ParityHostAgent.msi*, *ParityAgent6.0.2.msp* and *Parity6.0MaintenancePack.msp*) were not digitally signed. Patch 4 corrects this problem.
- Performance Issues on DFS Shares for Windows XP and Windows 2003
 - Details: In some instances, the Parity Agent shows degraded performance over DFS shares. A workaround is available in this release that improves performance in these cases. Contact Bit9 Support for information on how to enable this work-around.

- **Drift Report Generation Times Incorrectly Calculated**
 - Details: The time required to generate Drift Reports was incorrectly calculated in previous releases, leading to events indicating “Baseline drift report generation is slow,” when this was not really the case. In this release, the report generation time is correctly calculated and these misleading events are not generated.

Corrective Content in Parity 6.0.2 GA Release (Build 305, Patch 3)

- **Device Approvals Lost after Parity Upgrade**
 - Details: Devices that were approved in the Parity Console prior to Parity Upgrade could lose their approval status after upgrade. This release correctly retains device approvals created prior to upgrade.
- **Globally Approved Files Remain Pending on Endpoints**
 - Details: Occasionally, a Globally Approved file would remain Pending on some endpoints, while being approved elsewhere. This release correctly ensures that all endpoints receive all approvals.
- **Performance Issues with Per-User Rules**
 - Details: Application of large numbers of Custom Rules to multiple users could result in performance issues when the Parity Agent started. Performance in this case is improved in this release.
- **Timestamps on Digital Signatures not Treated as UTC**
 - Details: When the timestamp in a digital signature was checked, local time was used instead of UTC. This left a window in which a valid signature would be considered invalid. In this release, UTC is used in all time checks.

Corrective Content in Parity 6.0.2 GA Release (Build 300, Patch 2)

- **Certain Executable Packages not Marked as Installer**
 - Details: Certain executable installation packages were not properly recognized as installers. This release correctly identifies these executables.
- **Agent-Server SSL Communication Issue**
 - Details: In some circumstances, when the Parity Server sent a large amount of data to the Parity Agent, the underlying SSL implementation would send a small amount of corrupt data. This caused the Agent to fail to receive approvals and other data. In this release, the data are correctly formatted to avoid this problem.
- **Upgrade Failure with Large Rules**
 - Details: After an upgrade from a prior version of Parity, complex Custom Rules would cause the upgrade process to fail. This occurred when the new Parity Server completed post-upgrade processing during its first execution. This problem is corrected in this release.
- **Upgrade Failure on SQL Server 2005**
 - Details: When upgrading Parity installations that use SQL Server 2005, the Parity Server upgrade procedure failed to upgrade the underlying Parity schema. In this release, the schema is correctly upgraded.
- **Permission Issue with Daily Tasks**
 - Details: During processing of daily maintenance tasks, the Parity Server incorrectly assumed the presence of *ALTER SERVER* permissions on the associated SQL Server. This caused a harmless error message during processing. In this release, the maintenance tasks test for the *ALTER SERVER* permission and act accordingly.
- **Stop Error During Name Normalization**
 - Details: In very limited circumstances, an issue with name normalization would cause a Stop Error (also known as a Blue Screen). This release adds additional checks to ensure that this does not occur.
- **No Data from Meters**
 - Details: When Meters were added or modified through the Parity Console, the Parity Server was not correctly instructing Agents to send the data necessary for metering. This problem is corrected in this release.

Corrective Content in Parity 6.0.2 GA Release (Build 292, Patch 1)

- **Trusted User with Predefined Groups**
 - Details: Trusted Users with Predefined Groups (for example “Local Administrators”) did not function correctly when used in conjunction with UAC on Vista and above. In this release, Predefined Groups are functioning correctly as documented.
- **Excessive Growth of Temporary Database**
 - Details: A Parity Server background function could cause excessive growth in the SQL Server temporary database. This task is now performed in smaller increments, which limits the use of the temporary database.
- **Certain CAB Files not Scanned in Trusted Directory**
 - Details: Certain CAB files, notably those contained within Windows XP Service Packs or contained within Microsoft Office, were not properly extracted during Trusted Directory processing. This release correctly extracts and processes these CAB files.
- **Incorrect Slashes in Notifier Messages**
 - Details: When file names were reported in emails generated from the Notifier, the slashes contained in the path were incorrectly displayed as “/”. This release correctly uses “\” in these file names.
- **Occasional Stop Error for Reparse Points**
 - Details: In rare circumstances, while processing a file system reparse point, the Parity driver would encounter a Stop Error (also known as a Blue Screen). This release prevents this Stop Error from occurring.

Corrective Content in Parity 6.0.2 GA Release (Build 267)

- **Trusted User with Predefined Groups**
 - Details: Trusted Users with Predefined Groups (for example “Local Administrators”) did not function correctly. In this release, Predefined Groups are functioning correctly as documented.
- **Custom Rules with Predefined Groups**
 - Details: Custom Rules with Predefined Groups set in “User or Group” (for example “Authenticated Users”) did not function correctly. In this release, specifying a group in a Custom Rule works correctly.
- **Deadlock in the Parity Server**
 - Details: Under heavy load, the Parity Server could deadlock while initializing new threads. This issue is addressed in this release.

Corrective Content in Parity 6.0.2 GA Release (Build 264)

- **Parity Server initiated Parity Agent resynchronization frequently**
 - Details: The Parity Server initiated resynchronization of the Parity Agent in some circumstances. This release provides control over that resynchronization when deemed necessary by Bit9 Support.

Corrective Content in Parity 6.0.2 GA Release (Build 263)

- **Re-initialized or re-synchronized Parity Agents appear to be at 0% synchronization**
 - Details: When a Parity Agent was either re-initialized or instructed to re-synchronize its file information, it did not correctly inform the server of its progress. This caused the Parity Console to indicate 0% synchronization, even though the *Files on Computers* page contained all appropriate files. In this release, the Parity Agent sends correct progress information.
- **Incorrect mapping of process name in ArcSight CEF format**
 - Details: The process name for events in the CEF format was mapped to “DeviceProcessName” and not “deviceProcessName” as documented. The mapping has been corrected in this release.

- When the *Computer* field for a Trusted Directory is edited, computer information is lost
 - Details: When editing a Trusted Directory, the *Computer* field was editable. However, editing this field did not correctly transfer the Trusted Directory to the newly specified computer. This is fixed in this release.
- Approvals out of date and *CL Version* not incrementing on *Computers* page
 - Details: After an approval or other action that updated the *CL Version*, some systems remained at an earlier *CL Version*, showing “Approvals out of date,” while other systems were updated. In some circumstances, this issue was caused by a deadlock in the Parity Server. The deadlock is addressed in this release.

Corrective Content in Parity 6.0.2 LA Release (Build 258)

- *REG_EXPAND_SZ* keys not correctly expanded in custom rules
 - Details: The Parity Agent did not correctly expand registry keys of type *REG_EXPAND_SZ* when using “<Reg:...>” macros in custom rules. In this release, any unexpanded environment variables will be correctly expanded.
- Server identity reset during SQL Server outages prevented agent connection
 - Details: In very limited circumstances, SQL Server errors would cause the Parity Server to re-generate its internal identifying information. This prevented Parity Agents from connecting to the server. In this release the server will instead log an error and exit.
- Certain Inno Setup installers were not marked as such
 - Details: Certain Inno Setup installer executables were not flagged as installers by the Parity Agent or Trusted Directories. This problem has been addressed.

Corrective Content in Parity 6.0.2 LA Release (Build 255)

- Trusted User fails when using the Windows “runas” command
 - Details: An installer that was executed from a command prompt would not be correctly approved when the command prompt had been created using “runas”. This problem has been fixed.
- Parity Agents became disconnected with an HTTP 500 error
 - Details: In some circumstances, Parity Agents would be disconnected from the Parity Server and not reconnect until the Parity Server was restarted. This issue has been addressed.
- System Configuration page in Parity Console cleared all checkboxes
 - Details: If the System Configuration page was edited, but the edit was subsequently canceled, all checkboxes would be reset to “off”. In this release, the checkboxes retain the state they were in prior to editing.
- WebEx Trusted Updater not working properly
 - Details: Some WebEx updates were being blocked when the WebEx Trusted Updater was enabled. Several issues are addressed in this release to improve the performance of the WebEx Trusted Updater for all browser types.
- Additional settings for the Sophos Trusted Updater
 - Details: When deploying Parity in combination with Sophos, frequent Sophos updates can cause excessive events. Additional Parity configuration settings can address this issue. These settings are detailed in Solution 309 in the Customer Support Portal.
- Server does not correctly import multi-level certificates
 - Details: When importing a certificate for agent communications, the information displayed in the Parity Console is for an intermediate certificate and not the correct certificate. While this release does not directly address this issue, steps to remediate it are contained in Solution 312 available in the Customer Support Portal.
- Missing events for Custom Rule and Cache Consistency approvals
 - Details: Files approved via a Custom Rule or by the Cache Consistency Check did not generate approval events. These events were added in this release.
- Documentation missing for Policy Status
 - Details: Information was added to the “Using Parity” guide and online help detailing the Policy Status field in the Computer Details.

Corrective Content in Parity 6.0.2 LA Release (Build 247)

- Files blocked as Pending on DFS shares
 - Details: Under certain circumstances, Parity agents were not able to obtain the fully qualified name of a file located on a DFS share. These files would block as Pending and the event associated with the block would not have a hash. The problem was predominantly seen on Windows XP and is addressed in this release.
- Incorrect mapping of process name in ArcSight CEF format
 - Details: The process name for events in the CEF format was mapped to “destination process name” and not “device process name” as documented. The mapping has been corrected in this release.
- Additional parameters for performance tuning
 - Details: In situations that require high performance tolerances, it may be necessary to limit the tracking of specific process activity. Additional kernel configuration parameters were added in this release so that Bit9 Support can address performance issues where tolerances are tight.

Corrective Content in Parity 6.0.2 LA Release (Build 245)

- “Access Denied” error on upgrade from prior versions
 - Details: Upgrades to Parity 6.0.2 would fail in some circumstances with an error of “Access Denied” (code 5). This is addressed in this release, provided Parity agent approvals are up-to-date.
- Event type and sub-type were blank for computers deleted automatically
 - Details: Automatic deletion of an offline computer generated an event that did not contain an event type or sub-type. This event now has the same type and sub-type as the event generated when a computer is deleted by a Parity administrator.
- Baseline Drift alert triggers multiple events and sends multiple emails
 - Details: When an alert was configured for “Drift on All Computers”, multiple events and emails were sent when the alert condition was met. Now, a single event and email is generated when the alert condition is met.
- Windows Update blocks after upgrade
 - Details: Windows Update rules were not correctly propagated to Parity Agents after upgrade from previous 6.0 releases. This could cause blocks during Windows Update. Windows Update rules now propagate correctly when agents are upgraded.
- Memory information reported as zero
 - Details: In the Parity Console, the Computer Details page for 64-bit computers showed a Memory value of zero. Memory is now reported correctly for 64-bit computers.

Corrective Content in Parity 6.0.2 LA Release (Build 241)

- Windows Side-by-Side (SxS) errors could prevent agent installation
 - Details: In some cases where Windows Update(s) either weren't performed or had errors and didn't complete, Parity Agent could not be installed because of a missing assembly in Windows SxS. In this release, the agent will install even if there is an issue with SxS. A computer with this issue will still need to be rebooted.
- Agent upgrades could take longer than expected
 - Details: In previous releases, some operations during agent upgrade could increase the amount of time between deactivation of the old agent and activation of the new one. Once installed, the Parity 6.0.2 agent will require less time for upgrades, minimizing the amount of time that the agent is not running.
- Blocked files require action on both Parity and Windows dialogs
 - Details: When Parity blocks a file, the agent computer might display both a Parity Notifier and a Windows dialog. In this release, responding to the Notifier dismisses both the Notifier and the Windows dialog in more situations than in previous releases.

- **Custom Rules defaulted to wildcard**
 - Details: In past releases, if the default value (“*”) for paths and processes in a custom rule was accepted, a new rule could have wide-ranging effects. Now, if you choose a menu option that requires a path or process, there is no default. You must provide an explicit path or process, or an error message will appear.
- **Custom Rules were missing the Write Action “Allow”**
 - Details: “Allow” is a supported write action for a Custom Rule, but it did not appear on the write action menu on the Add/Edit Custom Rule page. This option is now available.
- **Wildcard use was limited in Parity rules**
 - Details: In previous versions of Parity, rules that allow entry of a path (Custom, Registry, and Memory rules) required that if *any* wildcards are used in a path, a wildcard must appear at the beginning or in the last element in order for the path to match any files, processes, or registry entries. In addition, only the asterisk (zero or more) wildcard was supported. Parity 6.0.2 lifts the restriction on wildcard location, and allows the use of both asterisk and question mark (one of any character) wildcards in any combination.
- **Trusted directory approvals could override user-specified file bans**
 - Details: Although file-specific bans should take precedence over the presence of a file in a trusted directory, that was not always the case in previous releases. Now bans will remain in effect even when a file is found in a trusted directory.
- **Approved files could lose approval status during consistency checks**
 - Details: Some files occasionally lost their “approved” state when an agent was carrying out internal consistency checks such as those initiated by the system administrator for diagnostic purposes. This condition has been eliminated.
- **Some large or complex installations on agent computers reported multiple groups**
 - Details: In previous releases, Parity reported several separate installer groups for some large or complex installations. Parity 6.0.2 improves the installation grouping information that is reported with these installations, increasing the percentage of files appearing as part of the expected parent installer group.
- **Some agent computer conditions caused inaccurate time stamps on files**
 - Details: In some cases, conditions on an agent computer caused timestamps for files to be inaccurate. This problem has been addressed.
- **Files marked as ROM images were not considered executable**
 - Details: Previous releases of Parity did not consider files marked as ROM images to be executable. Parity 6.0.2 extends security to cover ROM images and controls file execution according to the policies and rules in effect for the files on those images.
- **Performance issues with BigFix integration**
 - Details: Integration with Parity could have a performance impact on BigFix servers. Parity 6.0.2 includes improvements to address this.
- **Certain actions on agent computers caused performance issues**
 - Details: Some actions on an agent computer, such as deletion of large directory trees, could cause Parity to have a negative impact on system performance. Parity 6.0.2 includes many improvements to agent performance that reduces its impact in this situation.
- **Potential Cross-site scripting vulnerability**
 - Details: A potential cross-site scripting (XSS) vulnerability was discovered on the Parity Console. This vulnerability has been eliminated.
- **Performance optimization for certificate validation**
 - Details: In previous releases, the length of certificate validation times affected agent performance and could result in blocks. The performance of certificate validation for trusted publishers has been improved, and previously seen blocks due to excessive validation times have been eliminated.
- **Trusted Directory icon stayed green regardless of actual status**
 - Details: In previous releases, the status icon on the Software Rules/Directories page for a trusted directory remained green when the trusted directory was no longer valid. Now, when a trusted directory is invalid (e.g., because the agent on the computer where the directory located is uninstalled or deleted from Parity), the status indicator for the directory becomes red.

Corrective Content in Build 6.0.1.188

- Certain executables are incorrectly marked as installers
 - Symptoms: An executable is marked as an installer when that should not be the case.
 - Details: A non-installer component of the Microsoft .NET Framework version 4 was being marked as an installer. This release remediates this situation and correctly detects that these executables are not installers.
- Parity Reporter stops
 - Symptoms: The Parity Reporter service stops unexpectedly.
 - Details: When certain error conditions occurred, the Parity Reporter service was stopping instead of re-trying its connection to Parity Knowledge. Error handling while communicating with Parity Knowledge has been improved in this release to re-try correctly.

Corrective Content in Build 6.0.1.187

- 64-bit installers are not correctly processed in Trusted Directories
 - Symptoms: Placing a 64-bit installer in a Trusted Directory may not cause approval of its contents.
 - Details: Because the contents of 64-bit installers were not being properly expanded, the individual components of each installer were not being approved. This could cause execution blocks in some circumstances. In this release, 64-bit installers are correctly expanded.

Corrective Content in Build 6.0.1.186

- Parity Agents do not accept Timed SecCon Override codes
 - Symptoms: In some circumstances, an incorrect code would be generated for Timed SecCon Override.
 - Details: For override codes, Parity switched from using a word-based algorithm to one based on nonsense words in Parity 6.0.1. Codes for older agents were being generated using the wrong word list. This has been addressed in this release.
- Agents do not finish initializing or use a high percentage of CPU
 - Symptoms: Agents stay in initialization, or previously initialized agents consume a high percentage of CPU.
 - Details: Files of certain sizes caused a loop during analysis, preventing initialization or analysis from completing. This issue has been corrected in this release.
- Certain processes are incorrectly promoted
 - Symptoms: On Windows XP with the Windows Update Parity Updater enabled, processes that are not required for the update are occasionally promoted.
 - Details: The Windows Update mechanism runs through the Windows “service host.” Occasionally, processes not required by Windows Update that are running in or launched by the same service host will be promoted. This release addresses the issue.

Corrective Content in Build 6.0.1.185

- Agent Synchronization percentages remain low or zero
 - Symptoms: Synchronization percentages remain low or zero for many Agents, even after several weeks. This is particularly prevalent in large Parity deployments.
 - Details: Each Parity Agent determines whether it needs to resynchronize its file states with the Parity Server. In some cases, once it had notified the server of a full resynchronization, the Agent incorrectly pruned its internal list of items that need to be synchronized. This would cause a full resynchronization to be scheduled if the Agent went offline, even briefly. Existing Agents must be upgraded to this release in order to resolve this issue. Contact Bit9 Support for a workaround on the Parity Server if you are unable to upgrade Parity Agents in a timely fashion.

Corrective Content in Build 6.0.1.184

- **Performance issues copying large files on high latency networks**
 - Symptoms: Copying files to and from network shares was slower with earlier versions of Parity 6.0.1 compared to 6.0.0. Performance problems were especially noticeable over high latency network connections, such as those in remote offices.
 - Details: Parity monitors all network file access and will add nominal overhead to operations on these files. Earlier versions of Parity 6.0.1 increased the number of network requests related to file access monitoring by 15% over Parity 6.0.0, adding a noticeable amount of latency. This release reduces the network processing overhead so that it is less than it was in Parity 6.0.0.
- **Backlog of event processing and automatic restarts of Parity Server under high load**
 - Symptoms: Lower than expected Synchronization percentage across all Parity Agents and occasional automatic Parity Server restarts.
 - Details: A Parity Agent will request resynchronization of its file data with the Parity Server under certain conditions. This is expected behavior. This request can block for extended amount of time while clearing Parity Agent state in the Parity Server database. If this happens, the multithreaded environment on the Parity Server can allow the same Parity Agent can come back with a new request that temporarily blocks the main request processing thread. In previous releases, this caused the Parity Server deadlock detection to restart the Server. This no longer occurs in this release.
- **Interoperability issue with IBM Rational ClearCase**
 - Symptoms: All executable content in ClearCase dynamic views is blocked because the files cannot be opened for analysis.
 - Details: ClearCase devices appear as local paths despite having network file system characteristics, which leads Parity to misinterpret the name of the file, causing a block. This has been corrected in this release.
- **Interoperability issue with Entrust Entelligence Group Share**
 - Symptoms: Files block unexpectedly, and the Parity Notifier shows path names containing the string "eedmk".
 - Details: Group Share uses a "shadow volume" mechanism. In earlier versions of Parity 6.0.1, there were issues in the resolution of names of files opened via an NTFS file identifier rather than via a file name. This version of Parity 6.0.1 provides for correct name resolution; you must use Group Share version 2.3 to take advantage of this solution.
- **Dashboards are empty after an upgrade**
 - Symptoms: Parity Console Dashboards have no content after an upgrade to 6.0.1.
 - Details: The Parity Console was not correctly handling re-authentication of the logged in user's current session. This has been corrected in this release.
- **Trusted Directory fails to correctly identify certain file as installers**
 - Symptoms: An installation package is placed in a Trusted Directory, but not all files within it are approved. This occurs with certain Windows Updates (e.g. "WindowsXP-KB2485376-x86-ENU.exe").
 - Details: When Trusted Directory scanning identifies a file as an archive, it performs a preliminary determination of the archive type. In some cases, when the preliminary identification was later determined to be incorrect, the identification process was not restarted, leading to incorrect identification. This has been corrected in this release.
- **Software installation times are excessively slow**
 - Symptoms: Installations of certain software packages on computers running the Parity Agent are as much as five times slower as for non-Parity systems.
 - Details: The Parity Agent was not ignoring certain file read operations when no file handles were open. File names for files opened via an NTFS file identifier were excessively processed. This has been corrected in this release.
- **Certain USB devices do not appear in the Parity Console**
 - Symptoms: Some USB devices attached to computers running the Parity Agent did not appear in the Device Rules page on Parity Console.

- Details: When processing information on a newly inserted USB device, earlier versions of Parity incorrectly normalized some device names. This blocked access to detailed device information, which prevented reporting of the device to the Parity Server. Normalization has been improved in this release.

Corrective Content in Build 6.0.1.180

- Disabled periodic agent cache consistency checks.
- Added support for additional commands in **dascli**.
- Addressed a deadlock in the Parity Server under high load.

Corrective Content in Build 6.0.1.177

- Addressed an issue causing interactions with Symantec Endpoint Protection.
- Addressed a backward compatibility issue with “Allow and Promote” rules.
- Addressed a crash in the Parity Agent while analyzing certain files.
- Addressed a crash in the Parity Server under high load.

Corrective Content in Build 6.0.1.176

- Addressed an issue in which the Parity Server did not process all items in a **Trusted Directory** in a timely fashion.
- Addressed a memory leak in the Parity Server with high volumes of agent connects and disconnects in high load environments.

Corrective Content in Build 6.0.1.175

- Addressed an issue in which nested Microsoft Installer packages were not correctly approved in a **Trusted Directory**.
- Addressed agent performance problems with **DFS network shares**.
- Addressed an issue with **network printers**.
- Improved upgrade behavior.
- Addressed an issue with renaming directories.

Corrective Content in Build 6.0.1.162

- Addressed an issue in which, after a new Parity Server installation, a **Notifier dialog** appeared indicating that some System files were banned.
- Addressed an issue in which Parity was blocking **Java updates**.
- Addressed an issue in which **Trusted Directory status** on Parity Server was not being updated appropriately.
- Addressed an issue in which Parity Server was reporting a **SecCon change** for an agent when one had not taken place.
- Improved Parity Server **Alert processing**.
- Addressed an issue with Parity Agent in which **CPU consumption** for parity.exe would occasional spike and impact system performance.

Corrective Content in Parity 6.0.1 GA Release

Parity Rules

- Addressed macro case-sensitivity and drive letter expansions in **custom rules**.
- Addressed a problem processing **registry rules** with wildcard characters.
- Addressed delay in processing of changes to **tamper protection setting** by clients.
- Addressed an issue where rules specific to Parity 5.x clients were being sent to all clients when they were modified.
- Improved coverage of default performance rules (in **custom rules** list) for system files.

Agent Performance

- Improved agent initialization and cache consistency check performance and reduced CPU and I/O activity.
- Addressed agent performance problems on cloned or ghosted systems, and on clients deployed from a base image.

Agent Interoperability

- Improved compatibility with Windows 7 SP1 and Windows Server 2008 R2 SP1 beta.
- Improved interoperability with Microsoft Forefront and Security Essentials.
- Improved interoperability with Entrust Group Share 2.2 (requires Group Share 2.2 Patch #159093).
- Better support for Microsoft .NET applications in the .NET Updater.

Agent Security

- Added integrity checks on the client database.
- Improved detection of client configuration changes and client database consistency.

Other Corrective Content

- **Duplicate Client Handling:** Better handling of duplicate of client IDs on different endpoints; added the ability to force clients to reregister from the console.
- **Server Performance:** Improved server response to large amounts of new file information.
- **Upgrade Installations:** Addressed a problem restoring database backups over 2GB.

[Back to Table of Contents](#)

Known Issues and Limitations

- In rare cases, agent upgrades may be blocked because older Bit9 MSI or MSP packages referenced during upgrade have no global file state. This can occur after a server upgrade

from a release *prior to* 6.0.2.228, 7.0.0.1229, or 7.0.1.1109. If you have upgraded from a version prior to those listed, you may have this problem if:

- Users report that the Parity Notifier shows MSI or MSP blocks after you have enabled agent upgrades.
- On the console Events page, you notice multiple file block events for the same MSI or MSP files.
- Agents have an Upgrade Status of "Upgrade Scheduled" but do not ever change to "Up to Date" and have an Upgrade Error of "Agent Upgrade: Unknown error executing" or "Agent Upgrade: Failed executing".

If this situation occurs, do the following:

1. **Turn off automatic agent upgrades:** In the Parity Console, go to the **Administration > System Configuration** page and click on **Advanced Options**. On the Advanced Options tab click the Edit button at the bottom of the page, in the Parity Agent panel, choose Disabled on the menu, and then click the Update button at the bottom of the page.
2. **Locally or globally approve the Bit9 MSPs or MSIs that are blocking.**
3. **Turn automatic upgrades back on:** Follow the same procedure as step 1, except choose Enabled on the menu.

Note: If you are using a third-party software distribution method to upgrade agents, disable that distribution until you approve the blocking files.

- If you encounter this situation and are unsure of whether to approve the blocked files, contact Bit9 Technical Support.
- If you use the "Export to CSV File" feature in a Parity table (such as the Computers page), there is a limit of 25,000 on the number of rows that can be exported.
- Some or all memory rules are not supported on certain operating systems:
 - Memory rules are not supported on Windows Server 2003 64-bit.
 - Kernel Memory Access rules are supported only on computers running Windows XP or Windows Server 2003 without SP1.
 - Dynamic Code Execution rules are supported only on computers running 32-bit operating systems. On Windows XP SP2, if the system-wide DEP Policy is set to "AlwaysOff", dynamic code execution memory rules cannot be enforced, but Parity will report as though they were enforced. If the policy is set to "OptIn" (the default) or "OptOut", then these rules will be enforced on systems running XP SP2.

- In Memory Rules, do not use Prompt as the action for Dynamic Code Execution rules. This could cause a deadlock situation.
- If a Registry Rule is configured to block writing to a full path (no wildcard on the left), the rule will block attempts to rename and delete a key or value, but it will not block creation of a new key. However, no values can be created under this key.
- If a Parity Server has no file bans in effect, the first execution of a file on a computer whose agent is in the Monitor SecCon is not reported as a Parity event.
- Currently, internal CD and DVD drives are identified by Parity as removable drives, but they cannot be approved. Because of this, if you set the Write or Execute permissions in the Device Control Settings to either **Deny** or **Permit Approved** in a policy, these actions will be blocked in on internal CD and DVD drives in computers belonging to that policy. Contact Bit9 Technical Support for assistance if this issue poses a problem at your site.
- By default, computers running Microsoft Vista or Windows 7 operating systems have User Access Control (UAC) enabled. With UAC, users are not actually members of a built-in, privileged group unless they have been given "elevated privilege". Because of this, a Parity rule that relies on a pre-defined group to identify a user may not work for computers running Vista or Windows 7. If a group definition is necessary for a rule, consider using security groups you have defined rather than the pre-defined groups.
- In previous releases, missing assemblies in Windows SxS due to Windows Update failures could cause agent installation to fail. In Parity 6.0.2, agent installation succeeds in these cases, but the agent will not start until you reboot the computer, and since the agent is not running, it cannot report its status. If a system you expected to appear on the Computers page in Parity Console is not shown, check whether its agent is actually running and reboot if necessary.

[Back to Table of Contents](#)

Contacting Bit9 Support

For your convenience, Bit9 Technical Support offers several channels for resolving support questions:

Technical Support Contact Options
Web: www.bit9.com
E-mail: support@bit9.com
Phone: 877.248.9098 (877.BIT9.098)
Fax: 617.393.7499
Hours: 8 a.m. to 8 p.m. EST

Reporting Problems

When you call or e-mail Bit9 technical support, please provide the following information to the support representative:

Required Information	Description
Contact	Your name, company name, telephone number, and e-mail address
Product version	Product name (Parity Server, Parity Agent, or Parity Knowledge) and version number
Hardware configuration	Hardware configuration of the Parity Server or computer (processor, memory, and RAM)
Document version	For documentation issues, specify the version of the manual you are using. The date and version of the document appear after the copyright section of each manual.
Problem	Action causing the problem, error message returned, and event log output (as appropriate)
Problem severity	Critical, serious, minor, or enhancement

[Back to Table of Contents](#)