TOP 3 WORKSTATION LOGS TO MONITOR

Improve endpoint security by monitoring your security, Sysmon and PowerShell logs

Written by Brian Hymer, solutions architect, Quest

Quest

Introduction

WHY FOCUS ON WORKSTATION SECURITY?

Most attacks today begin on user workstations. Why? Well, in part it's because workstations, unlike servers, are typically the province of non-technical users, who are easier prey for attackers. For example, Verizon's 2017 Data Breach Investigations Report (DBIR) found that 1 in 14 users were tricked into following a malicious link or opening an infected attachment — and a quarter of those were duped more than once. Workstation users also fall victim to drive-by downloads from websites they think they can trust, unwittingly insert USB drives containing ransomware or other malware, and make other critical mistakes that allow attackers to gain a foothold in the corporate network.

It's easy to lay all the blame on users, but attacks are getting more sophisticated all the time. In particular, hackers mine social media for information they can use to make their phishing emails increasingly convincing, and hide malware in downloadable files that seem to be the genuine assets that users are looking for. Even IT pros themselves sometimes fall for these more sophisticated gambits.

The other side of the coin is that users' workstations are also particularly vulnerable, for several reasons. First, many modern exploits rely on interactive local user actions, which are the most common scenario on workstations. Second, most attacks exploit malicious file and web content — and workstations come into contact with far more files from the Internet than servers do. Finally, many vulnerabilities involve third-party GUI-driven applications used on workstations, such as internet browser extensions, and keeping these applications properly patched remains a weak point in many organizations.

Most attacks begin on user workstations. Learn how to fight back.

HOW CAN YOU GET THE INFORMATION YOU NEED?

This combination of non-technical users and vulnerable workstations is irresistible to hackers, so you have to make endpoint security a priority. The key to catching attacks as early as possible and stopping them before real damage is done is to properly monitor your workstations. But what's the best way to do that? This ebook reveals the three most important logs for tighter Windows workstation security — the security, Sysmon and PowerShell logs — and details exactly which events to collect for each and why.

Of course, with the high number of workstations and large amount of log data at most organizations, there are some real challenges involved in collecting and archiving workstation logs, let alone monitoring, searching and analyzing them. So we'll also briefly show how Quest® InTrust® and Quest® IT Security Search can help you further strengthen endpoint security without running your IT teams ragged or blowing your storage budget.

Quest

Window security log

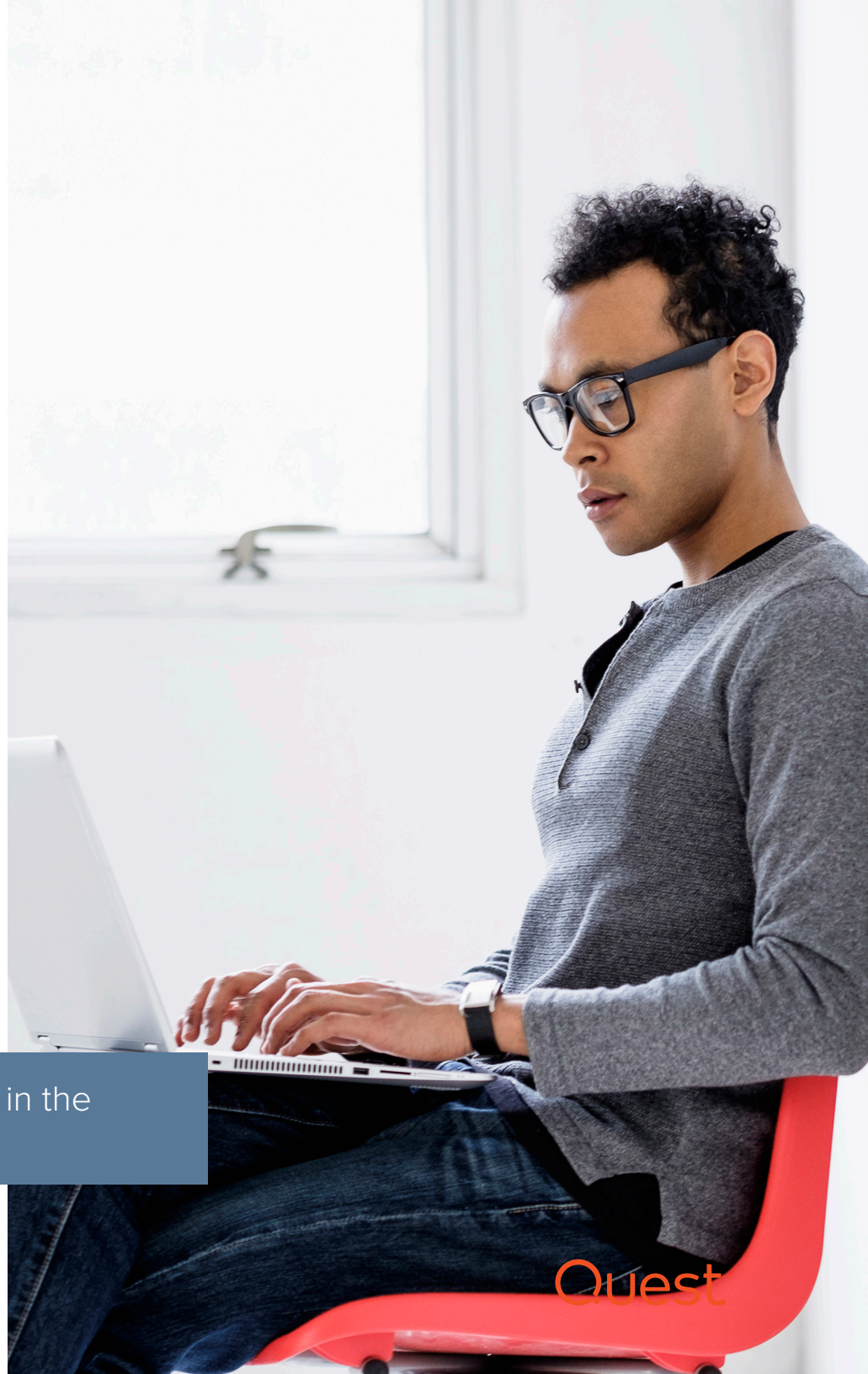
Effective workstation log monitoring begins with the Windows security log. It's the main record of security-related activity on workstations, and many important security events are logged only there.

WHICH EVENTS TO COLLECT

The Windows security log is the only place to get the following events:

- **Local user and group enumeration (events 4798 and 4799) —** Malicious code often enumerates the local user accounts and local groups on the workstation to find useful credentials. There are some legitimate reasons for enumerating users and groups, so you will get some false positives. But if you baseline those cases, monitoring events 4798 and 4799 can help you spot malicious code before it can move laterally to other systems and use the credentials it has harvested.
- **Local account creation and group changes (events 4720, 4722–4726, 4738, 4740, 4767, 4780, 4781, 4794, 5376 and 5377) —** Attackers also often create or modify local accounts and local groups (especially the Local Administrators group), so you want to keep an eye on these events.
- **Logon attempts with local accounts (event 4624) —** Users normally log on to their workstations using a domain account, so successful

Many important security events are recorded only in the Windows security log.



and failed attempts to log in using a local account can be a great indicator of attacks. Event 4624 logs every type of logon attempt, including domain logons, but it's easy to filter those out because in those cases the domain name is the computer name.

- **Logon with explicit credentials (event 4648) —** This event is generated when a process attempts to log on by explicitly specifying another account's credentials. This occurs legitimately with scheduled tasks or when using the "RUNAS" command, for example. But since most scheduled tasks are not run on workstations, this event can also indicate a malicious process trying to start another process with specific credentials or an attacker mapping a drive to another computer using credentials they've collected.
- **When was the user physically present and active (events 4800—4803)** — Remember that users sometimes stay logged on for weeks at a time, so in addition to looking at logon and logoff events, you also need to look at when the workstation console was locked and unlocked. Any activity on a workstation while it's locked demands further investigation.
- **Firewall configuration change (events 4944–4958) —** Depending on how the system is set up, applications can automatically add exceptions to the Windows firewall as they're being installed, particularly when the user has local admin authority. Those exceptions don't have to be deliberately malicious to create serious security gaps. If you rely on the Windows firewall as a significant security control, you need to know about any changes to its configuration.
- **Plug-and-play device connections (event 6416, Windows 10 only) —** Malware often enters a workstation through USB drives or other plug-and-play devices. It's important to audit connections from all such devices — for example, there have been attacks through keyboards, as well as USB drives that show up as keyboards.

There are other important events that you can get from the security log — but I recommend getting them from Sysmon instead because the quality of the data is better. These events include:

- Process creation
- Network connections
- Registry changes
- File creation

We'll explore these events in the "Sysmon" section below.

> Expert tip: Create honey objects — folders or SharePoint document libraries with tempting names — and watch for attempts to access them.

HOW TO COLLECT EVENTS IN THE SECURITY LOG

A workstation's audit policy determines which type of information about the system you'll find in the security log. Windows uses nine audit policy categories and 50 audit policy subcategories to give you granular control over which information is logged. You can choose whether to log successful events, failed events or both.

I recommend turning on auditing for the following audit subcategories. Choose both "success" and "failure" for Logon events, and "success" only for the others.

Logon/Logoff

- Logon
- Logoff
- Account Lockout
- Other Logon/Logoff Events

Account Management

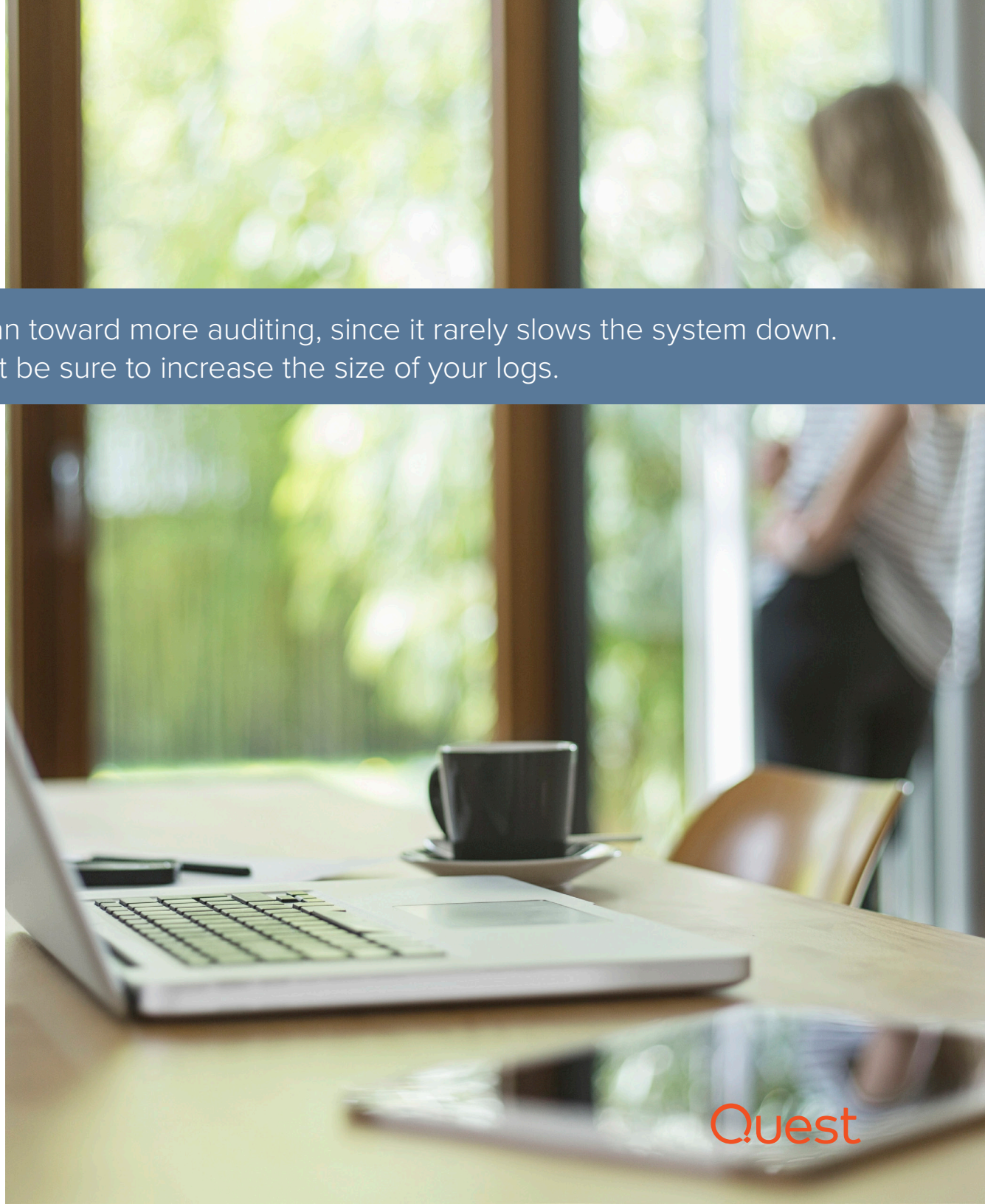
- User Account Management
- Security Group Management

Policy Change

- Audit Policy Change
- Authentication Policy Change
- Authorization Policy Change

This list represents the minimum auditing I recommend; you might well want to turn on additional subcategories. Remember, you don't have to collect and archive every event that's logged. I always lean towards more auditing because it doesn't slow the system down (except in a very few cases, such as auditing all access to all files). Just be sure to increase the size of your logs; I recommend something between 200MB and 1GB.

Lean toward more auditing, since it rarely slows the system down. Just be sure to increase the size of your logs.



Sysmon

Sysmon is a free service from Microsoft that monitors system activity and records it in a Windows event log, which is also called “Sysmon.”

WHICH EVENTS TO COLLECT

As noted earlier, there is some overlap between the Sysmon log and the security log. I recommend using Sysmon for the following events because the quality of the data is better:

- **Process creation (event ID 1) —** The Windows security log will tell you when an EXE process is started and provides its name and path. But attackers can easily create a malicious program with the same name as a legitimate tool, such as c:\windows\notepad.exe, or modify an existing program to perform illicit actions. To catch those, you need a hash of the file’s contents, which the security log does not provide but Sysmon does. A hash is a

There is some overlap between the Sysmon log and the security log, but for certain events, Sysmon provides better quality data.

unique mathematical digest of the bit stream of the file, so replacing or altering the file results in a different hash. By using free web applications to analyze the hash, you can easily determine whether a file has been changed or replaced with known malicious code, such as a Trojan app.

- **Network connections (event ID 3) —** Monitoring network connections can also help you spot attackers. Of course, the volume of data will be very high, so you'll need to establish baselines of normal activity to spot suspicious connections. Both the security log and Sysmon enable you to collect network connection events, but Sysmon links each connection to a process through the ProcessID and ProcessGUID fields, and also provides the source and destination host names IP addresses, port numbers and IPv6 status.
- **Registry changes (event IDs 12–14) —** Once attackers get their malicious code on a workstation through a phishing email, drive-by download or other avenue, they want that code to run even after the workstation is rebooted. The most common way of achieving that persistence is to modify the registry, for example, by adding a run key. You can track this with the Windows security log, but Sysmon provides far more context, including critical details like who made the change, which computer they used, when it happened, the process ID, and the new name of any key or value that was renamed.
- **File creation (event ID 11) —** The Windows security log will tell you that a new file was created (or overwritten) in a certain folder, but it does not provide the name of that new file. Sysmon does, which makes it easier to identify and investigate suspicious file creation events so you can block attacks sooner. In particular, you should monitor autostart locations like the Startup folder, as well as temporary and download directories, where malware often appears during initial infection.

I also recommend using Sysmon to monitor the following events that are not included in the security log:

- **Driver and image loads (event IDs 6 and 7) —** In addition to providing the hash for EXE process starts as explained above, Sysmon also tracks DLL and device driver loads, which attackers also use. It even reports on whether the file is signed, who signed it and whether the signature is valid.
- **Remote thread creation (event ID 8) —** Sophisticated hacking tools can inject a DLL into a running process and then fire up that DLL on another thread. While this capability has legitimate uses, such as debugging, you need to know when it happens on a production system.
- **Raw access reads (event ID 9) —** The RawAccessRead event detects when a process conducts reading operations from the drive using the "\\.\" denotation. This technique is often used by malware for data exfiltration of files that are locked for reading, as well as to avoid file access auditing tools. The event indicates the source process and target device.
- **Named pipe creation and connection (event IDs 17 and 18) —** Some malicious software communicates with different components via a communications channel in Windows called named pipes.
- **WMI event activity (event ID 19) —** Malware can execute by registering a Windows Management Instrumentation (WMI) event filter. This event logs the WMI namespace, filter name and filter expression.

- **Named file stream creation (event ID 15) —** This event helps you detect malware variants that drop their executables or configuration settings via browser downloads.
- **File creation time change (event ID 2) —** To evade some types of file integrity monitoring, attackers can alter file creation times. For example, they will create a file but immediately change its creation time so it won't appear on a list of recently created files. Sysmon will catch that.

HOW TO ENABLE LOGGING

To install Sysmon, simply download the executable from Microsoft and then run the following command:

```
Sysmon -i
```

To control which event IDs are logged, specify the appropriate settings in an XML configuration file and then run this command.

```
Sysmon -c config.xml
```

Many of the events I listed above will generate some false positives, so, for example, you could specify a list of EXE files that Sysmon should exclude from monitoring. For a configuration file template that is a great starting point for system change monitoring, visit https://github.com/SwiftOnSecurity/sysmon-config.

Your Sysmon config file controls which events are logged.

PROTECTING SYSMON FROM TAMPERING

It's much easier to tamper with Sysmon than the Windows security log. Fortunately, there are a few techniques that can mitigate this risk. First, you can obscure Sysmon by changing the driver name (use the DriverName tag in the config file) and changing the service name (rename the executable before installing it).

However, there are still methods hackers can use to find Sysmon, so you also need to monitor for attacks on it. Fortunately, Sysmon tracks changes to itself using the following events:

- **Event ID 4 —** Reports on a Sysmon service state change (start or stop)
- **Event ID 16 —** Reports on a Sysmon config state change, including the hash of the config file

You can also set up a scheduled task via Group Policy that periodically wakes up and runs the following command to enforce the proper configuration (replace "server\share" with the correct path in your environment):

```
\\server\share\sysmon -i -accepteula -c \\server\
share\sysmon.xml

if errorlevel 1 \\server\share\sysmon -c \\server\
share\sysmon.xml
```

There are also third-party tools that can automatically reset your Sysmon configuration after improper changes.

PowerShell logs

WHICH EVENTS TO COLLECT

Hackers love to use PowerShell because it's so powerful. Therefore, it is critical to keep a close eye on PowerShell activity. There are two PowerShell logs: The Microsoft-Windows-PowerShell/Operational log gets most of the attention, but there is also the Windows PowerShell log. I recommend monitoring certain events from both of them:

Windows PowerShell log

- **Providers loaded (event ID 600) —** PowerShell providers are programs that make the data in a given data store available in PowerShell so that you can view and manage it. For instance, built-in providers include Environment (for managing Windows environment variables) and Registry (for managing the Windows registry). You can create your own providers and install providers that others develop. Whitelist the providers normally used in your environment to minimize false positives, and watch for new areas of PowerShell being used, which could indicate malicious activity. In particular, if you see that the WSMan

provider was loaded, you'll know that a remote PowerShell session has been started.

Microsoft-Windows-PowerShell/Operational (renamed to "Microsoft-Windows-PowerShellCore/Operational" in PowerShell 6)

- **Module logging (event ID 4103) —** Module logging provides more detailed auditing that includes every command executed and all of its parameters (but not the output of the command).
- **Script block logging (event ID 4104) —** Script block logging shows every block of PowerShell code that was executed, which provides a lot more context than seeing each individual command. Even if a hacker tries to hide or obfuscate a command, this event will show the actual PowerShell command that was executed, so it's far more powerful than capturing commands executed in the system. Also, this log can capture some of the low-level API calls being executed, providing even more details about what hackers are doing. This event is normally logged as Verbose, but if Microsoft detects that a suspicious command or scripting technique is being used in a block of code, it will be logged as a Warning instead.

HOW TO ENABLE LOGGING

The Windows PowerShell log captures events by default, so you don't need to enable logging to see which providers have been loaded (event ID 600).

You can turn module logging and script block logging on or off using the corresponding Group Policy settings under Administrative Templates | Windows Components | Windows PowerShell.

Hackers love to use PowerShell because it's so powerful.

Quest

Drawbacks to using these native logs

Together, the Windows security log, Sysmon and the PowerShell logs can give you some visibility into your user workstations that can help you spot and thwart attacks. But there are multiple reasons not to rely on just these logs and the native tools to ensure workstation security.

It's difficult and slow.

First there's the challenge of simply getting the logs from all your workstations in a timely and efficient way, especially if many of them are mobile laptops. Then you have to have workflows for analyzing them quickly and efficiently, so you can spot and investigate suspicious activity in time to prevent serious damage.

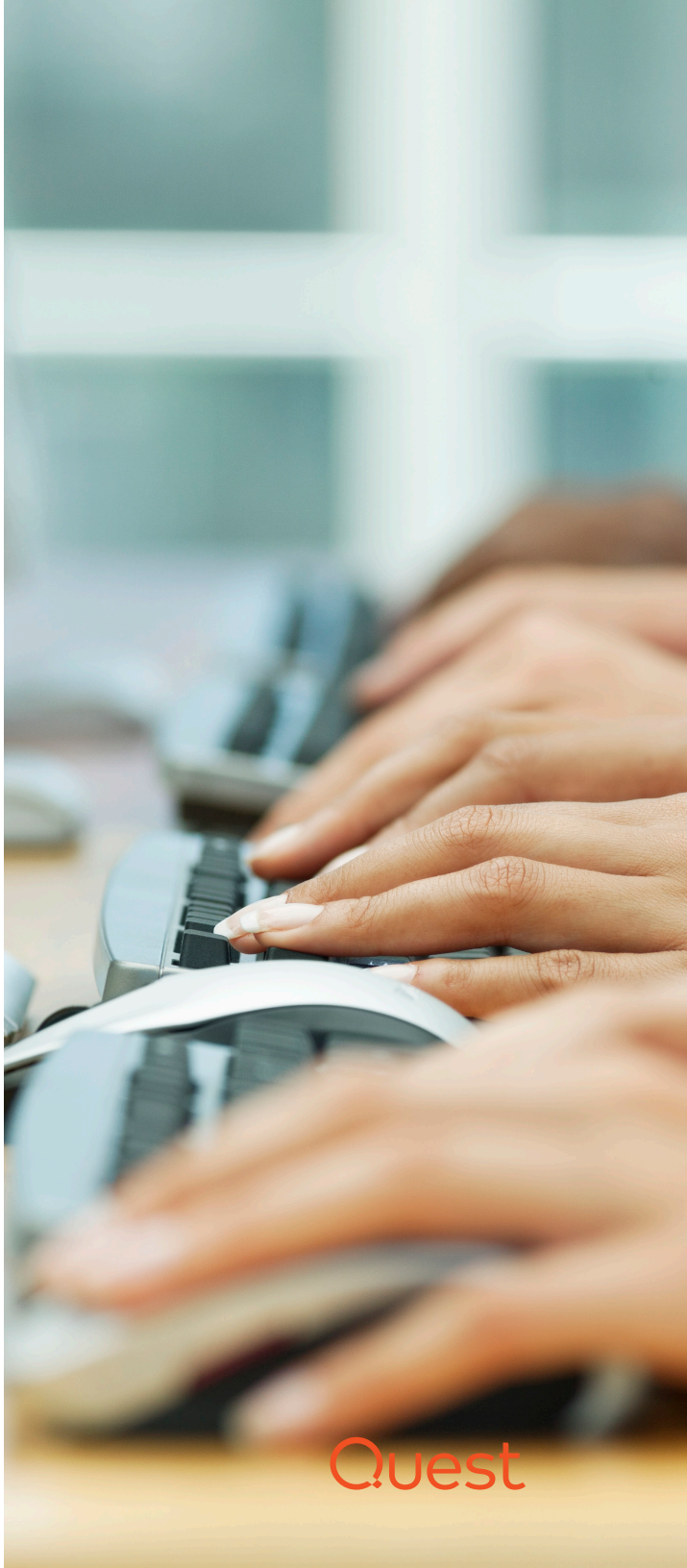
That's tough, in part because native logs are notoriously cryptic. For instance, earlier in this ebook I recommended enabling the Authorization Policy Change subcategory in the Windows security log. Doing so will enable you to see when permissions on files and folders are changed – but unless you're a complete Window log guru, you won't know much else, because the log data is not provided in anything close to a human-readable format. Instead, each time a permissions change occurs, you'll have to manually run a PowerShell command like this:

```
(get-acl <folder name>).access | ft
IdentityReference,FileSystemRights,AccessControlType,IsInherited,
InheritanceFlags -auto
```

In the time it takes you to spot the change to permissions and run this command, an attacker can easily slip into your network.

It's expensive in multiple ways.

Then there are the expenses. It takes time to perform all these log data management and analysis tasks, which drives up staffing costs. If you're using a security information and event management (SIEM) solution or log management tool that charges for processing based on events per second or megabytes per day, the high volume of events that native logging generates can run into serious



money. And you might be spending a bundle on storage as well, especially if you're storing the data without compression.

> With native tools, it's a challenge to even collect logs from all your workstations and laptops, let alone analyze them effectively.

You simply can't get the visibility you need.

Finally, there are the gaping holes. To start with, manual tasks by their very nature are ripe for human errors and oversights, which mean that you might well miss critical events that happen on your workstations. Moreover, the log data itself is incomplete and fragmented. For example, I explained how you can use Sysmon to monitor file creation events, but of course, file creation is just a tiny part of the larger task of file system auditing. Unfortunately, there is really no way to perform quality file system auditing — or many other critical tasks — using native tools.

Quest

Improving workstation security with Quest InTrust and IT Security Search

The right third-party tools can be a win-win — enabling you to dramatically improve workstation security while driving down personnel and storage costs. Together, Quest® InTrust® and IT Security Search give you the visibility you need into workstation activity, in an integrated, easy-to-use solution.

INTRUST

Quest InTrust is an event log management solution that enables you to collect, store, search and analyze massive amounts of IT data from numerous data sources, systems and devices — securely and efficiently. The data repository is indexed for fast searches and offers perhaps the best compression on the market: 20–1 with indexing and 40–1 without.

Even better, you don't have to be a Windows log expert to get actionable information from

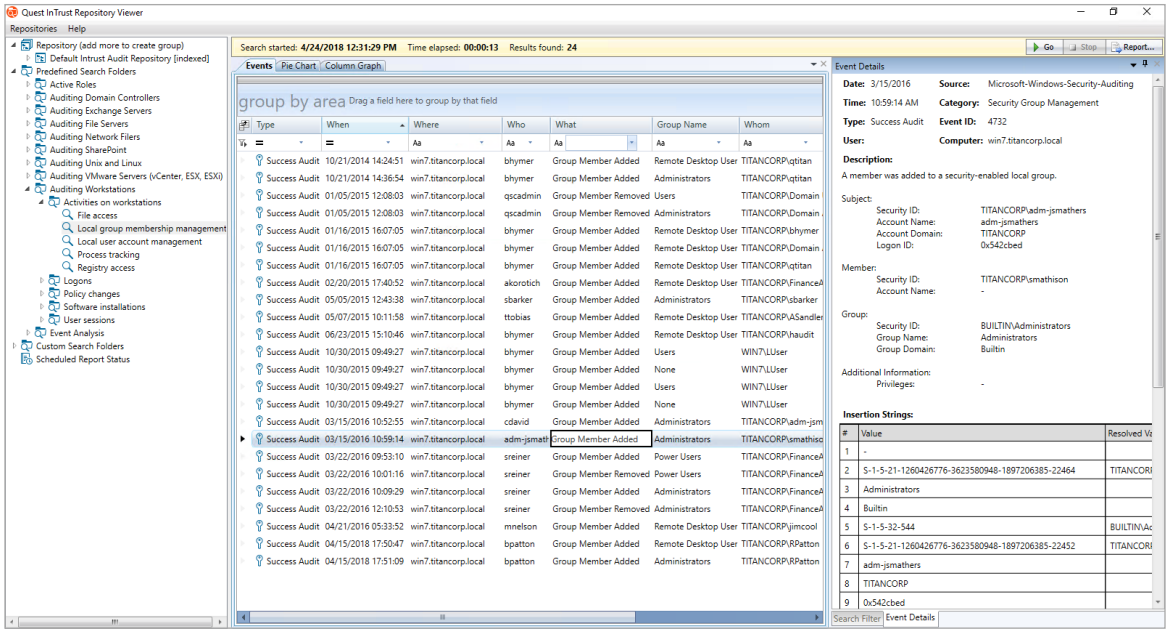


Figure 1. InTrust simplifies event log management to enable better workstation security while reducing costs.

You don't have to be a Windows log expert to get actionable information from InTrust.

Quest

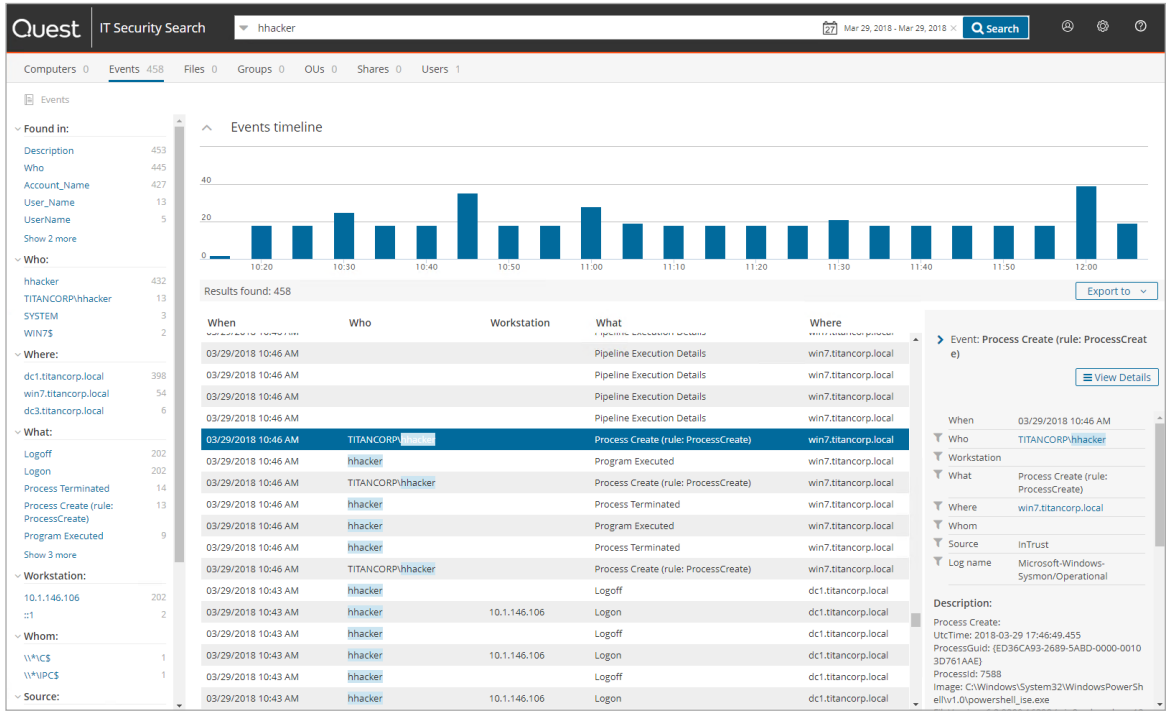


Figure 2. Reviewing Windows workstation log data in IT Security Search

IT Security Search enables fast security incident response and forensic analysis across disparate systems from a single pane of glass.

InTrust, because it normalizes critical Who, When, What, Where, Where and from Whom data into human-readable form, as illustrated in the built-in report shown in Figure 1.

On top of all that, InTrust offers both easy deployment and massive scalability: You can deploy InTrust agents to 5,000 workstations in just 20 minutes. InTrust can ingest roughly 60,000 events per second, and one InTrust server can monitor well over 10,000 endpoints. If you have more endpoints, all you have to do is add another InTrust server and split the load.

IT SECURITY SEARCH

Quest IT Security Search is a free solution included with InTrust, as well as with other Quest auditing solutions, including Enterprise Reporter, Change Auditor, Recovery Manager for AD and Active Roles. It pulls data from disparate IT systems and devices into a single pane of glass and provides a Google-like, web-based interactive search engine for fast security incident response and forensic analysis — with no training or log expertise required.

In particular, you can easily review the critical Windows workstation logs we just discussed. You can filter the data to eliminate noise, and dynamically pivot your investigation as other details emerge, as illustrated in Figure 2.

Quest

Conclusion

Attackers are constantly upping their game, and workstations are often their target of choice. Judicious use of the security, Sysmon and PowerShell logs can help you spot and block attacks, including ransomware and other malware, in time to prevent serious damage. And Quest InTrust and IT Security Search will simplify the process of collecting and analyzing that log data, along with lots of other data from across your environment, to slash response time, IT workload and storage costs.

To learn more, check out these resources:

- **In Trust —** quest.com/products/intrust
- **IT Security Search —** quest.com/products/it-security-search

ABOUT THE AUTHOR

Brian Hymer is a solutions architect at Quest and an expert on the Windows security log and Active Directory forest recovery. His 30 years of experience in the IT industry spans multiple sectors, including power, retail, healthcare, insurance and finance. During his 18 years at Quest, he has focused on helping customers around the globe implement and use Quest products in a wide variety of environments. He has also presented numerous worldwide webinars.

Quest InTrust and IT Security Search help you slash response time, IT workload and storage costs.

ABOUT QUEST

At Quest, our purpose is to solve complex problems with simple solutions. We accomplish this with a philosophy focused on great products, great service and an overall goal of being simple to do business with. Our vision is to deliver technology that eliminates the need to choose between efficiency and effectiveness, which means you and your organization can spend less time on IT administration and more time on business innovation.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (www.quest.com) for regional and international office information.

Ebook-Top3WorkstationLogs-InTrust-US-GM-33820

Quest