RESEARCH BLOG

Research by:

Aly Anwar (@alyanwarr)

EDRs decrease your enterprise security, unless properly hardened

August 24, 2022

Key take-aways

- **Default credentials** gave us full access to a popular Cynet 360 EDR system, its privileges, and functions
- In addition, we discovered **three high-severity vulnerabilities in the EDR,** arising from **weak access control on API endpoints**

Introduction

The use of Endpoint Detection and Response is growing rapidly and for good reasons. Their promise is to keep systems safe by monitoring and detecting malicious activities through data analytics and deception. After detection, EDRs also try to stop attacks – often through automated responses. EDRs usually need broad access to the system, and this is what can turn them into an interesting attack surface for hackers.

EDRs have created security issues before. Research by Optiv and others shows how EDRs by different vendors can be hacked through hooking. What these approaches have in common with other EDR related vulnerabilities and bypassing techniques documented in the National Vulnerability Database (NVD) – is that they focus on bypassing the detection of the EDR itself rather than abusing its functionalities and privileges. Our new findings highlight how **the overall security of an EDR can be compromised by a web interface interacting with the EDR**.

Our research found new configuration and design issues in a popular EDR system. During a recent red team exercise, we found **three now patched zero-day vulnerabilities in the Cynet 360 web portal.** We worked through a responsible disclosure process with Cynet, we successfully addressed the design issues in the latest Cynet 360 version. Now it is up to you to install the latest version and to address the default password issues.

This post revisits **how we identified these high-severity vulnerabilities, how they could have been exploited, and what EDR users and vendors need to do to stay safe.**

The application

Cynet 360 is an **EDR that allows users to monitor and respond to security threats.** Like other EDR solutions, Cynet 360 runs an agent on the target machines that continuously collects data and sends it to a master node where alerts are generated.

The default Cynet 360 setup also includes a web portal running on port 8443 for the management node, which will be the focus of this blogpost. The portal can be configured to be reached externally, internally, or limited to a local host network interface. The default configuration is having port 8443 accessible internally for the required communication between the agents, slave, and master nodes.

- **Collect:** handling scanning-related operations, e.g., running a scan, viewing results, specifying scope
- **Analyze:** security intelligence and behavior inspection operations
- **Alert:** viewing and redirecting alerts to the responsible entities
- **Remediate:** acting on the reported alerts via cleaning-up the infected files, isolating machines, running commands, etc.
- **Deception:** handling various deception techniques, e.g., decoy users

Additional details can be found in the Cynet 360 user manual.

The hacker’s wish list: five attack scenarios

We found five attack scenarios to be particularly interesting from an offensive point of view during red teaming:

#1 Execute commands on monitored endpoints

#2 Disable/redirect alerts

#3 Find out which users are decoy users

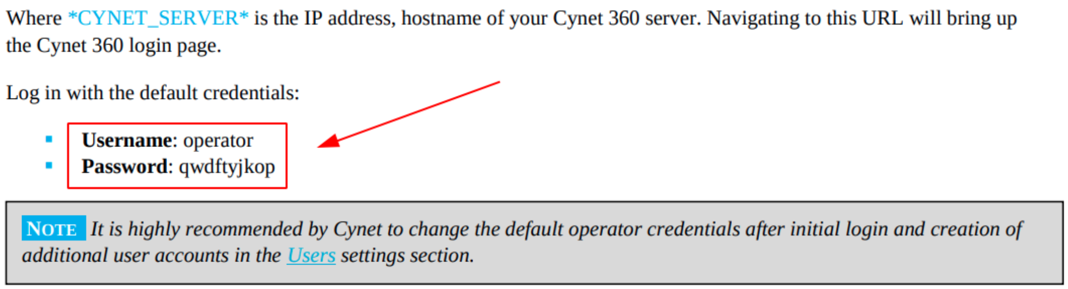
#4 Find out which exclusion profiles are in place

#5 Find out which monitoring profiles are in place

A default password issue allows hackers to achieve all five attack objectives. Three separate and new vulnerabilities allowed hackers to achieve three out of five.

The first wish: Grant me a default password

As a standard step in red teaming, we try to find **default credentials** for target software. To our surprise, **Cynet 360 has a default credential for the on-premise versions**, documented in the public user guide.



Where *CYNET_SERVER* is the IP address, hostname of your Cynet 360 server. Navigating to this URL will bring up the Cynet 360 login page.

Log in with the default credentials:

- **Username**: operator
- **Password**: qwdftyjkop

NOTE *It is highly recommended by Cynet to change the default operator credentials after initial login and creation of additional user accounts in the Users settings section.*

Figure 1: Default credentials listed in Cynet's 360 user guide

Users are encouraged to change the credentials. And yet, it worked **on all the Cynet 360 on-premise portals** we encountered during red team exercises.

Figure 2: Successful login to the Cynet 360 portal as operator

The ***operator*** account has all permissions to one or several Cynet servers – a goldmine for hackers. What can the hacker do?

Plain and simple: **Everything.** Let us explore two scenarios:

#1 Execute commands on monitored endpoints (Offensive remediation)

At this point all the possible blue team **remediation** actions could be converted into **offensive** actions. The hacker can run arbitrary commands on the machines, isolate, un-isolate, pull files, or shutdown/restart them. See Figure 3 for available options.

Figure 3: Available remediation options

#2 Disable/redirect alerts

If a hacker can disable or redirect the alerts, then they can cause as much or as little noise as they want without getting detected. Figure 4 pictures some of the available alert actions that we could access and as you can see in Figure 5, we were also able to modify alert recipients.

Figure 4: Multiple alert

Figure 5: Available options to change the alerts recipients

This checks off #1 and #2 from the hacker wish list. Attack scenarios #3, #4, and #5 can also be achieved, this time even without any default credentials.

The hacker's wish list – revisited

While it is certainly interesting what a hacker can do once able to login to the Cynet portal, it is more interesting what they can achieve **without any authentication.**

#3 Show me the Decoy users [CVE-2022-27969]

Implanting decoy users is a deception technique commonly observed in our red team exercises. Cynet 360 allows users to use a ***Decoy Users*** functionality that implants invalid credential information in the Windows Credential Manager. This data can later be accessed by other applications as saved credentials. Whenever a login attempt with decoy user credentials is submitted, Cynet 360 triggers an alert.

When doing a mass password spray or brute force attack, it is usually challenging from the red team/hacker's perspective not to touch a decoy user account.

During our unauthenticated analysis on the Cynet 360 web application, we found that the endpoint responsible for returning the list of decoy users is **missing access control**.

This means that **a remote unauthenticated hacker can make a request to the endpoint and find all the decoy users in the network.**

The endpoint can be reached at:

https://<cynet-server>:8443/WebApp/DeceptionUser/GetAllDeceptionUsers

Figure 6: Successful response from the Decoy Users endpoint while unauthenticated

the hacker can then avoid touching them and continue the attack path without triggering an alert.

#4 Anything that you are not monitoring? [CVE-2022-27967]

Sometimes we see environments in which the company introduces exclusion policies to the EDR. These exclusion policies enforce that the company trusts specific items (processes, directories, files, or extensions) and therefore there is no need for the EDR to monitor or scan them.

Cynet 360 allows users to specify such configuration through an ***Exclusion Profile***. We found that the endpoint responsible for listing such exclusion profiles is indeed missing access control and therefore **can reveal to an unauthenticated user what exclusion profiles are in place.**

The endpoint can be reached at:

*https://<cynet-server>:8443/WebApp/SettingsExclusion/GetExclusionsProfile?**profileId=1***

Figure 7: Successful response from the Exclusion Profiles endpoint while unauthenticated

Note that the *profileId* parameter is an incremental integer value.

This vulnerability becomes handy for a hacker, because they might be able to spot those specific directories, files, and processes that are not being monitored, therefore green light for an abuse.

For example, if a hacker reveals a custom exclusion profile for PowerShell, they can then use *powershell.exe* for executing malware, enumeration scripts, etc. without causing an alert.

By default, Cynet does not exclude *powershell.exe* from monitoring, however we witnessed such custom exclusion profiles in our redteam excercises.

#5 What are you actually monitoring? [CVE-2022-27968]

Likewise, Cynet 360 gives the user the ability to specify what file extensions they want to monitor through the ***File Monitoring*** functionality.

Like the previous two vulnerabilities, the file monitoring endpoint is missing access control. Therefore, a remote unauthenticated hacker can make a GET request directly to the endpoint and **reveal what**

Figure 8: Successful response from the File Monitor Profiles endpoint while unauthenticated

In this case a hacker can see that the ***Best Practice*** policy is the one in place. From the Cynet 360 user guide this entitles that the following file extensions are being monitored.

Figure 9: The Best Practice profile extensions as per the Cynet 360 user guide

From an attacking perspective, this is an important piece of information to know. An interesting attack to try at this point might be implanting malware with an extension that is not listed in the currently applied file monitoring profile, thus bypassing the forensics alerts/detections.

The Fixes

For users:
If you are running **Cynet 360 with version >= 4.5.6** you should be **safe against these three now patched zero-days**. If not, you should update your version as soon as possible.

In general, the mitigation advice here is **to always treat any security solution in your network as a possible target for hackers** and thus **take the necessary steps to audit/secure it.** These actions would typically include:

- Never keep accounts with default/weak passwords, especially administrative ones
- Restrict management interfaces to localhost IP of the running machine/server
- Fine-tune the privileges of dashboard users/operators according to the least-privilege concept
- Integrate monitoring/detection capabilities for unusual application access
- Conduct regular red teaming to uncover potential weaknesses in your environment

For EDR vendors, a fix for of such vulnerabilities would include:

- Enforce a change on the default operator password once the application has been deployed. Better yet: Ship with a randomly generated password

EDRs can be beneficial to detect and respond to attacks happening in your network, but when misconfigured or otherwise vulnerable, they can also impose a significant threat since they are generally granted high privileges.

This research was done by the SRLabs Red Team and first presented at HITB 2022 Singapore by Karsten Nohl and Jorge Giménez.

Editing by: Maria Bühner

RESEARCH BLOG

RECOMMENDED

RECOMMENDED

RECOMMENDED

Your Blockchain is only as secure as the application on top of it

Applications interacting with blockchain networks can be an attack surface to malicious actors and therefore need to be reviewed thoroughly.

EDRs decrease your enterprise security, unless properly hardened

Through default credentials we gained full access to the popular Cynet 360 EDR system, its privileges, and functions. Additionally, we discovered three high-severity vulnerabilities in the EDR, arising from weak access control on API endpoints

Smarter is not always wiser: How we hacked a smart payment terminal

Smart payment terminals can be vulnerable due to their expanded capabilities and the use of end-of-life OS, which result in a large attack surface. This post shares our experience hacking a popular smart point of sale device during an SRLabs retreat