CEH v12 Lesson 7 : Implementing Malware Concepts

Learning Outcomes

In this module, you will complete the following exercises:

- Exercise 1 — Malware and its Types
- Exercise 2 — Advanced Persistent Threat
- Exercise 3 — Antimalware Programs
- Exercise 4 — Malware Analysis & Countermeasures

After completing this module, you will be able to:

- Create a Fork Bomb as a Simple Virus
- Scan Using Windows Defender
- Using an Online Anti-Malware Scanner
- Use SUPERAntiSpyware

After completing this module, you have further knowledge of:

- Types of Malware
- Command and Control
- Use of Malware
- Components of Malware
- Advanced Persistent Threat (APT)
- Attributes of APTs
- Lifecycle of APTs
- The objective of Malware Analysis
- Types of Malware Analysis

- Malware Detection Methods
- Malware Countermeasures

Lab Duration
It will take approximately **1 hour** to complete this lab.

Exercise 1 — Types of Malware
Malware is a type of software designed to perform malicious activities that cause damage to the computer system or network. Malware is further divided into several categories based on the activities they perform.

In this exercise, you will learn about malware and various types of malware. You will also create a fork bomb.

Learning Outcomes
After completing this exercise, you will be able to:

- Create a Fork Bomb as a Simple Virus

After completing this exercise, you will have further knowledge of:

- Types of Malware
- Command and Control
- Use of Malware
- Components of Malware

Types of Malware
People often use the terms' virus and malware interchangeably; however, they are not the same. Malware is a category of malicious software containing different types, such as viruses, worms, and trojans.

Each type of malware has a set of distinct characteristics and features. For example, a trojan is malware hidden inside regular software, whereas a worm is designed to replicate itself over a network.

Malware can spread and infect in various ways. Some of the common methods are:

- Free applications or software, such as software cracks or pirated software
- Free file-sharing services, such as torrents or peer-to-peer
- Removable media
- An email that contains a malicious attachment

Some of the key types of malware are:

VirusA virus is a software program designed to cause changes in how a particular program behaves. A virus does this by changing the code of the target software program, which it gains access to by exploiting a security vulnerability. Due to the virus, undesirable changes can take place in its behavior.**Worm**A worm can be delivered by email or other network systems. For example, if a user receives an email with an attached worm file, opening the attachment will cause the worm to activate. Once triggered, a worm replicates itself from one system to the other(s) on the network. Eventually, it attempts to infect all systems on the network. The spread of a worm can cause network performance issues.**Trojan**A trojan is malware that provides unauthorized access to a victim's machine by posing as regular software. Often, a trojan is an attachment sent over email. As the email receiver executes the attachment, it connects back to the attacker's system. The attacker can then escalate privilege to perform administrative activities and lateral movements.**Keylogger**Keyloggers are tools that log user activity by capturing keystrokes, collecting screenshots, and recording application windows opened by a user. This tool is important for user surveillance and useful for law enforcement. When used maliciously, keyloggers can be a dangerous tool for secretly recording user system activity with the intent of harming the unsuspecting user. For example, the user's passwords can be logged by capturing keystrokes.**Spyware**

Spyware is designed for watching a user's behavior and activity on the system. Spyware can get into a system through an infected application or a standalone spyware program and enter a user's system through cookies. Cookies are stored on the user's machine, downloaded after visiting a website, and have certain privileges, like accessing browser settings and storing user data.

A malicious cookie could take advantage of these privileges to collect data and send the gathered data back to the attacker. Spyware quietly performs its tasks of stealing data, monitoring users' actions and activities, and gathering sensitive information. After it

gathers various types of information from an infected system, it can relay the information to the attackers.

Ransomware

Ransomware is malware that holds the user's files or system for ransom. The software does not cause any damage to the computer system or the network by itself; instead, it targets data by blocking or encrypting the user's access. The attacker demands a ransom (usually money in the form of non-traceable currency) from the user to decrypt or release the data.

The data can only be decrypted if the user has the decryption key, but there is no guarantee that the attacker will provide the decryption key after receiving the ransom.

Backdoor

A backdoor is access to an application that a developer creates. Generally, developers create a backdoor to remotely administer or gain access to an application. It can also be used to give the developers access in case they forget the administrative credentials. Users of these applications do not know that there is a backdoor.

Backdoors can also be created by malicious applications or malware. Attackers use these backdoors to access the system without letting a user know.

Logic BombA logic bomb works on predefined conditions, such as time or date. It may exist in the system but triggers only when the predefined conditions are met. For example, you can create a small script executed at a specific date and time. Once executed, it can cause severe damage, such as deleting key partitions in the operating system and data.**Crypto Malware**

Crypto malware is similar to ransomware. However, ransomware is openly used to demand money from the target. On the other hand, Crypto malware remains undetected and exists on the system quietly. It remains undetected to the extent that the user never gets to know anything about it, even sometimes undetectable by anti-malware tools.

Crypto malware is not used to steal data but to perform cryptomining, solving complex problems to make cryptocurrency. With the process of cryptomining, there is a certain amount of payoff to the attacker.

RootkitSome sensitive areas of computer software, such as an operating system, are not accessible to ordinary users. These are named root areas since they hold fundamental and essential modules of a software system. Software designed to gain unauthorized access to this root area is a rootkit. Getting access to these areas allows the intruder to perform harmful activities. For example, modifying the software structure and gaining unauthorized functions can damage the system.**Potentially Unwanted Program (PUP)**

People often mistake Potentially Unwanted Programs (PUPs) with malware. However, PUPs are not designed to perform malicious activity on a system. They are simply unwanted programs. For example, when you install an application from the Internet, you may notice that you have a new web browser toolbar added. This toolbar is the PUP.

Often, a PUP is bundled with the legitimate software you downloaded. They are also mentioned in the End User Agreement. However, most users do not read the agreement and rush through the installation, not realizing that they are installing a PUP along with the software.

PUPs can impact the system's performance. For example, they may run in the background and consume system resources, impacting system performance. Also, since you unknowingly installed these applications, they may have exploitable vulnerabilities that you are not aware of.

Fileless Virus

Unlike traditional malware, a fileless virus does not depend on a file executed from a hard drive. Instead, the malware directly operates from memory. When the file is loaded into the memory, it uses a scripting engine like Windows PowerShell to trigger the payload. However, there is no physical file being downloaded onto the hard drive.

Because there is no physical file involved, it is difficult for anti-malware or anti-virus applications to detect it. Also, the malware is being executed within a legitimate application such as PowerShell, so it is hard to stop the malicious script execution.

Command and Control

The behavior of malware can be directed and controlled by a Command and Control (C&C) server. When malware exfiltrates data, it is sent to this server. As well as directing and controlling malware, the C&C server can also control a botnet, and one C&C server can control and manage thousands of bots at any time. These botnets can operate over

architectures, client-servers, as well as peer-to-peer connections. Botnets are used for carrying out various types of attacks, such as distributed denial-of-service (DDoS) attacks, data theft, and spamming.

Use of Malware

Malware can be designed to meet a specific objective, and every malware developer may have a different objective. Let's look at some of the objectives that malware is designed.

- Steal confidential, sensitive, or personal information
- Infect and use one or more systems as a botnet
- Infect Web browsers to track user's Web usage
- Degrade system performance
- Delete information from a system

Malware must enter a target system to infect it. It could enter via various methods, such as:

- Rogue application
- Pirated applications
- Removable devices, such as USB
- File sharing services, such as NetBIOS and SMB
- Email attachments
- Vulnerabilities exploitation

After it is designed, malware needs to be delivered to the target to cause infection. For infection to occur, the malware must get inside the target and get triggered. The malware developer can use various methods to ensure the malware is delivered to the target system. Some of these delivery methods are:

- **Social engineering — click-jacking**: tricks a user to click on a link. When the link is clicked, malware is triggered in the background without the user's knowledge.
- **Spear phishing website**: it looks like a real website but is used to steal user credentials. Users are lured using social engineering to visit the website, and when

they enter user credentials, the malware is triggered.

- **Malvertising**: embeds malware-infected advertisements that look like real ones
- **Infected websites**: are used for spreading malware. The legitimate websites are infected without their owner's knowledge.
- **Drive-by-download**: is software that gets downloaded without the user's knowledge. The malware exploits the vulnerabilities in a web browser, and when a user visits a specific website, malware is downloaded onto a system.
- **SPAM**: are emails that contain malware as attachments or malicious code in the email body.

Components of Malware

Malware is designed with various components where each component has to perform a specific task. Even though there are various components, but not necessarily, an attacker would use all of them. Let's look at some key components:

- **Crypter**: is a code that prevents malware from reverse engineering and getting caught by a security program, such as IDS or IPS.
- **Downloader**: is a Trojan downloaded to the victim's system to provide access to the system.
- **Dropper**: is like a courier guy who drops the malicious code into the system. At first, the legitimate-looking program needs to be installed, which eventually executes the dropper that further releases the malicious code.
- **Exploit**: is a code that takes advantage of a vulnerability and exploits it.
- **Injector**: is responsible for injecting the malicious code into a vulnerable application.
- **Malicious Code:** this is the code behind the malware. It contains various commands that are triggered to conduct an attack.
- **Obfuscator**: is responsible for hiding the malicious code to prevent detection.
- **Packer**: provides the compression to make further the malicious code unreadable.
- **Payload**: is designed to perform a specific activity when activated in a system.

Task 1 — Create a Fork Bomb as a Simple Virus

A fork bomb is a simple virus that affects Windows machines. When executed, it goes in an infinite loop, so it does not stop. Eventually, the system hangs and crashes because of the shortage of system resources. You can design the virus to reach many different objectives.

For example, you can create a small virus to delete a Windows operating system's System32 directory files. On execution, this code can damage the system, and it may require extensive time and skill to fix it:

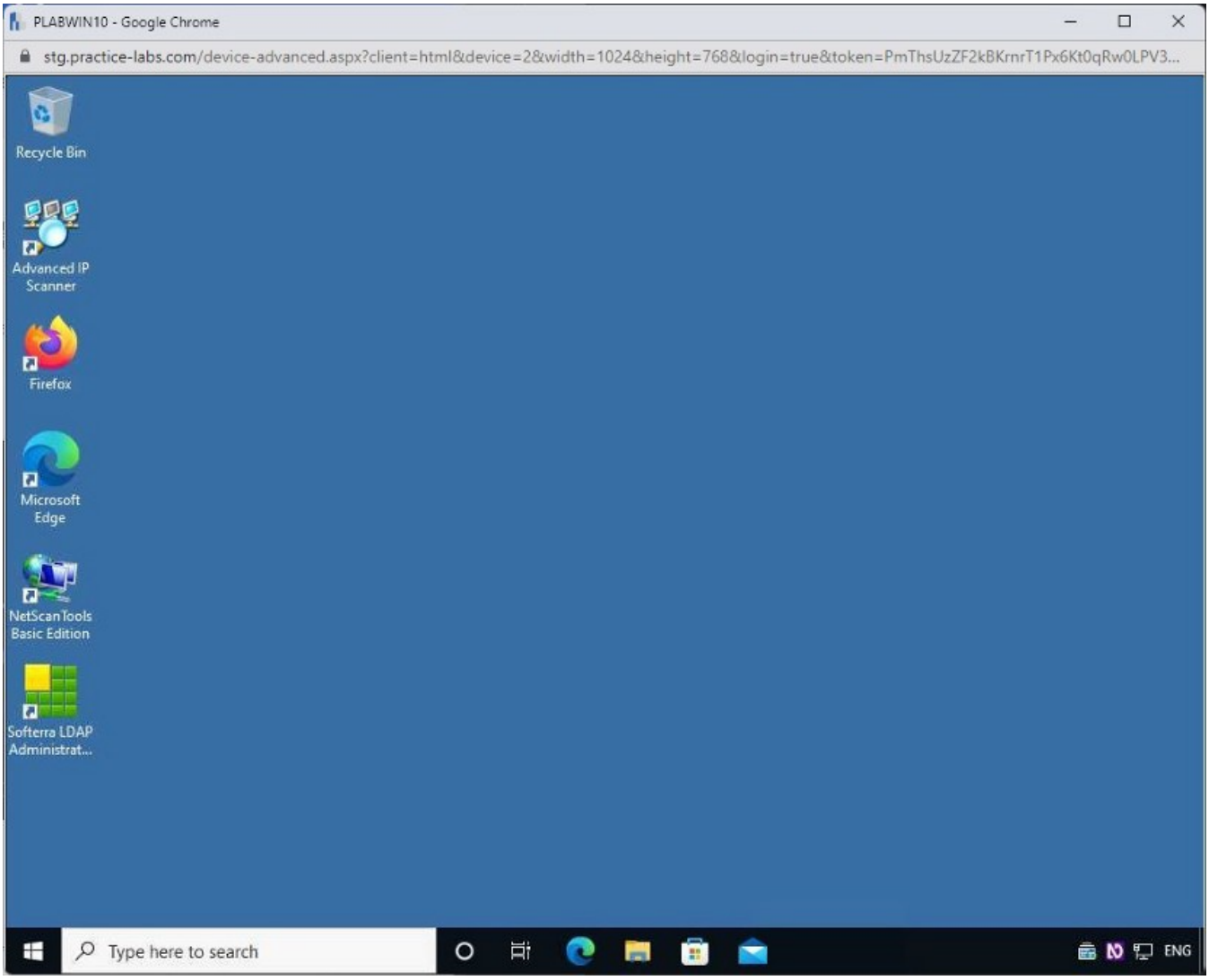
```
@echo off
Del c:\windows\system32\*.*
Del c:\windows\*.*
```

The @echo off command does not show the command being executed, allowing it to run in the background.

In this task, you will create a fork bomb using a batch file and execute it.

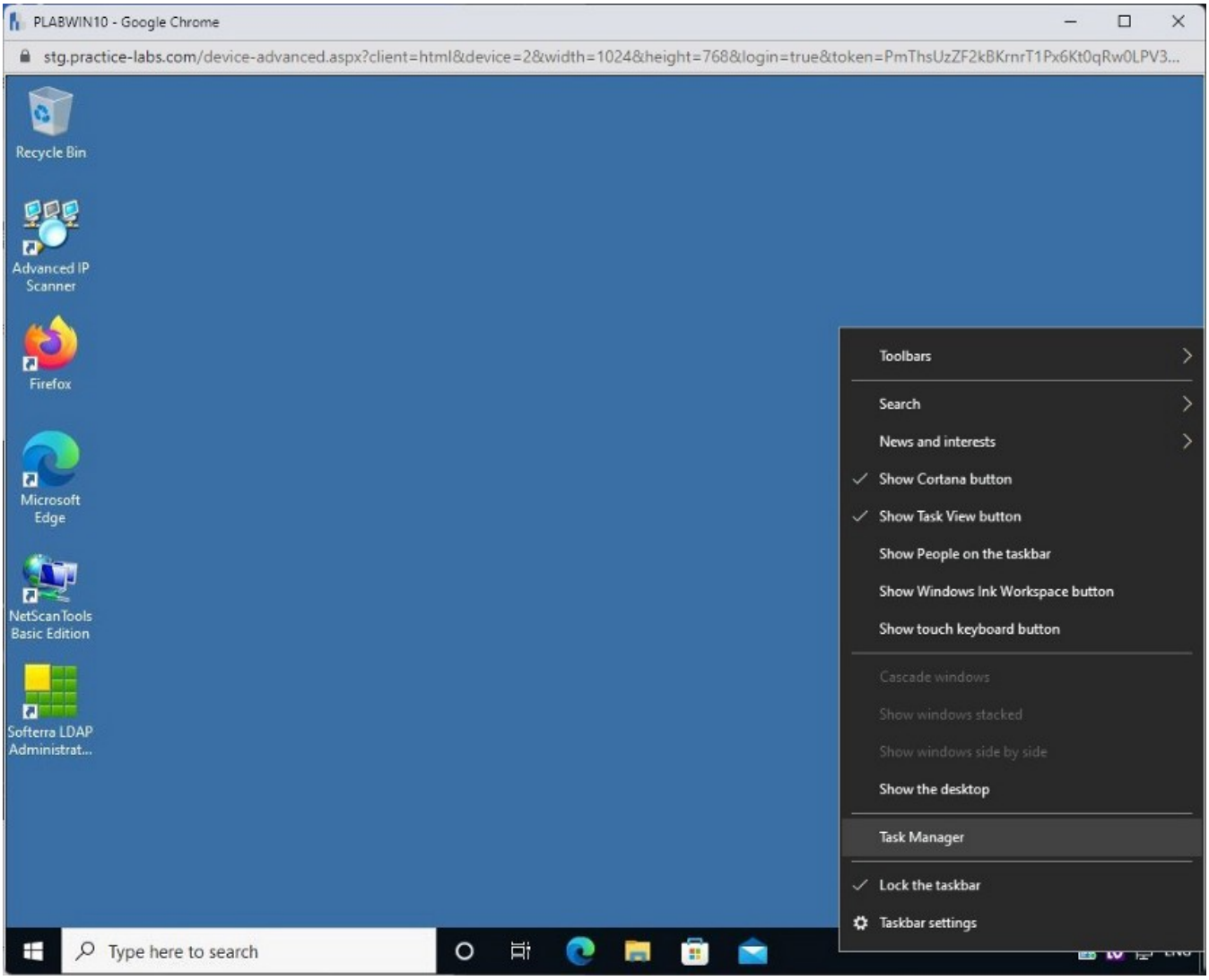
Step 1

Ensure you have powered on the required devices and connect to **PLABWIN10**.



Step 2

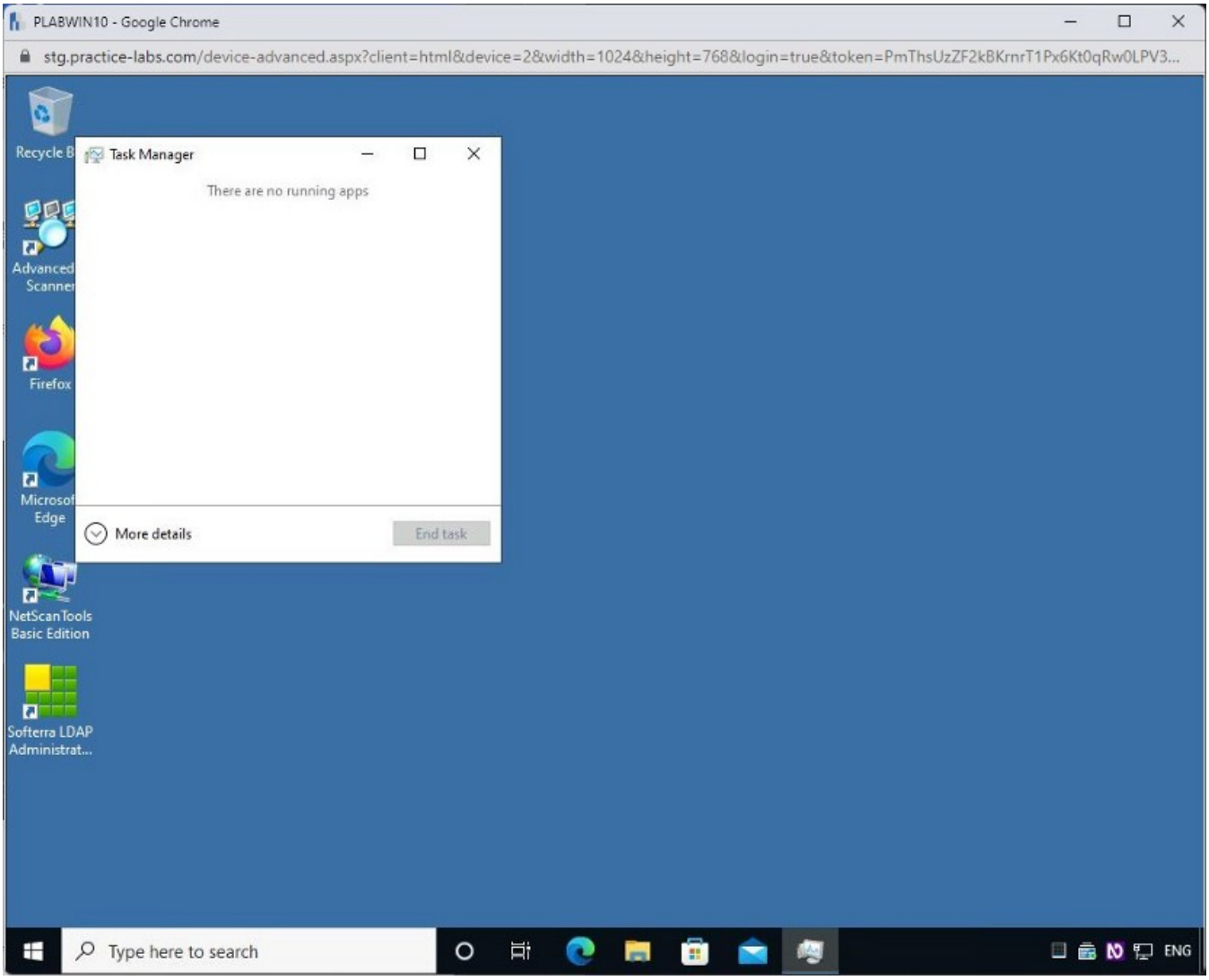
To open **Task Manager** from the **PLABWIN10** desktop, right-click the taskbar and select **Task Manager**.



Step 3

The **Task Manager** window is displayed.

Click the **More details** drop-down arrow.

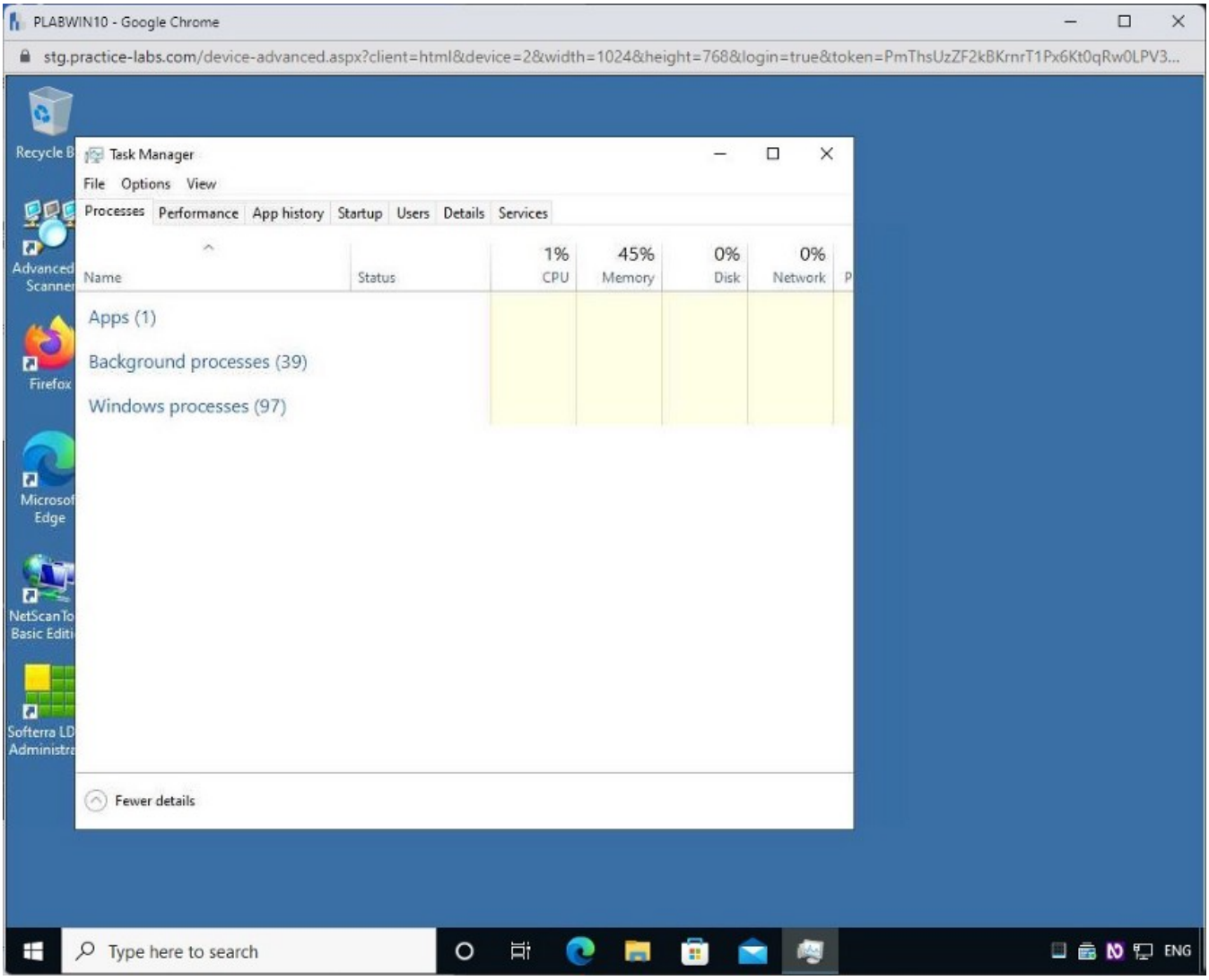


Step 4

The **Task Manager** window expands with the **Processes** tab selected by default.

Click the **Performance** tab.

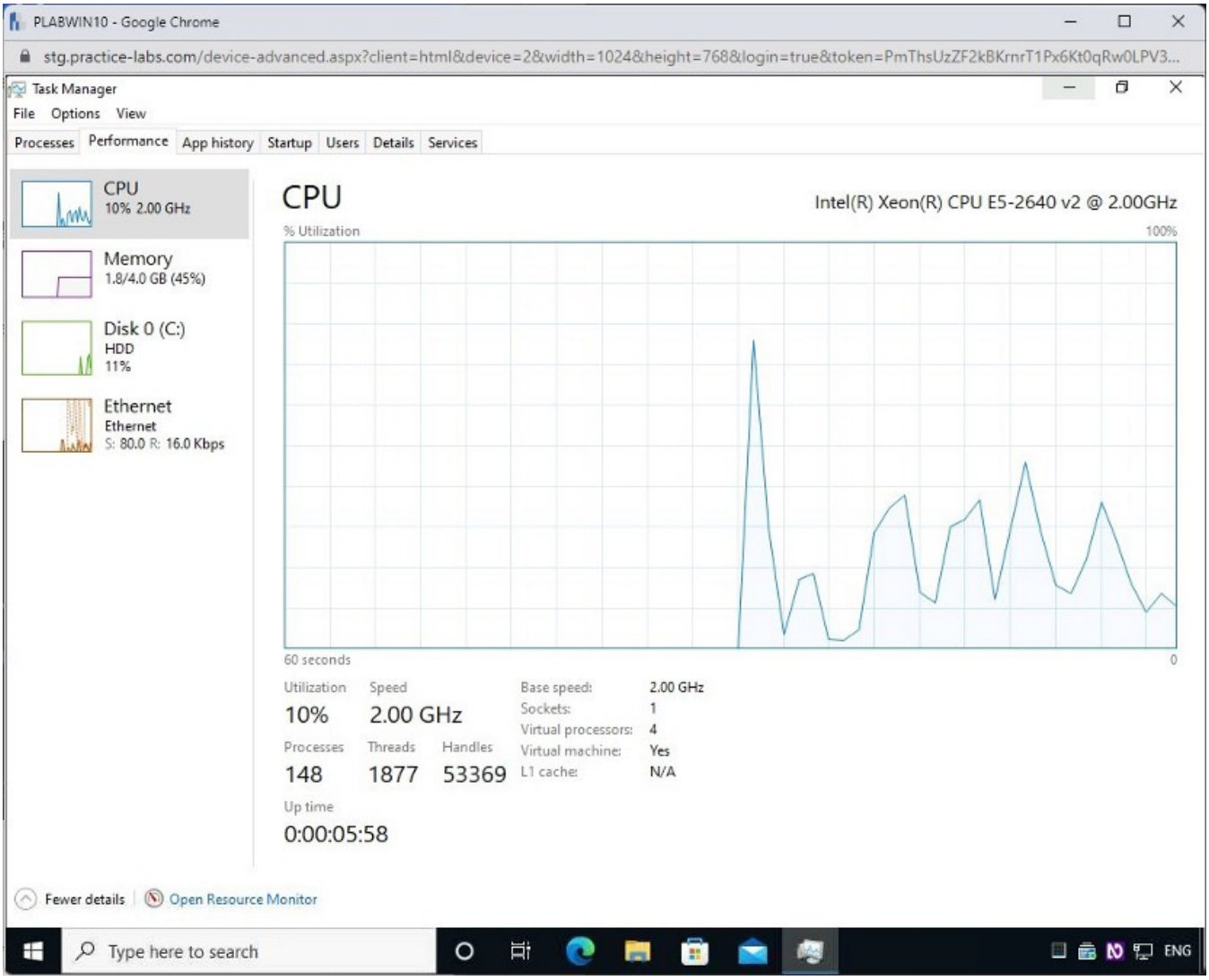
Note: *The Performance tab will show you the fork bomb in action.*



Step 5

Notice that the performance of various components, such as **CPU** and **Memory,** is displayed on this tab.

Minimize the **Task Manager** window.



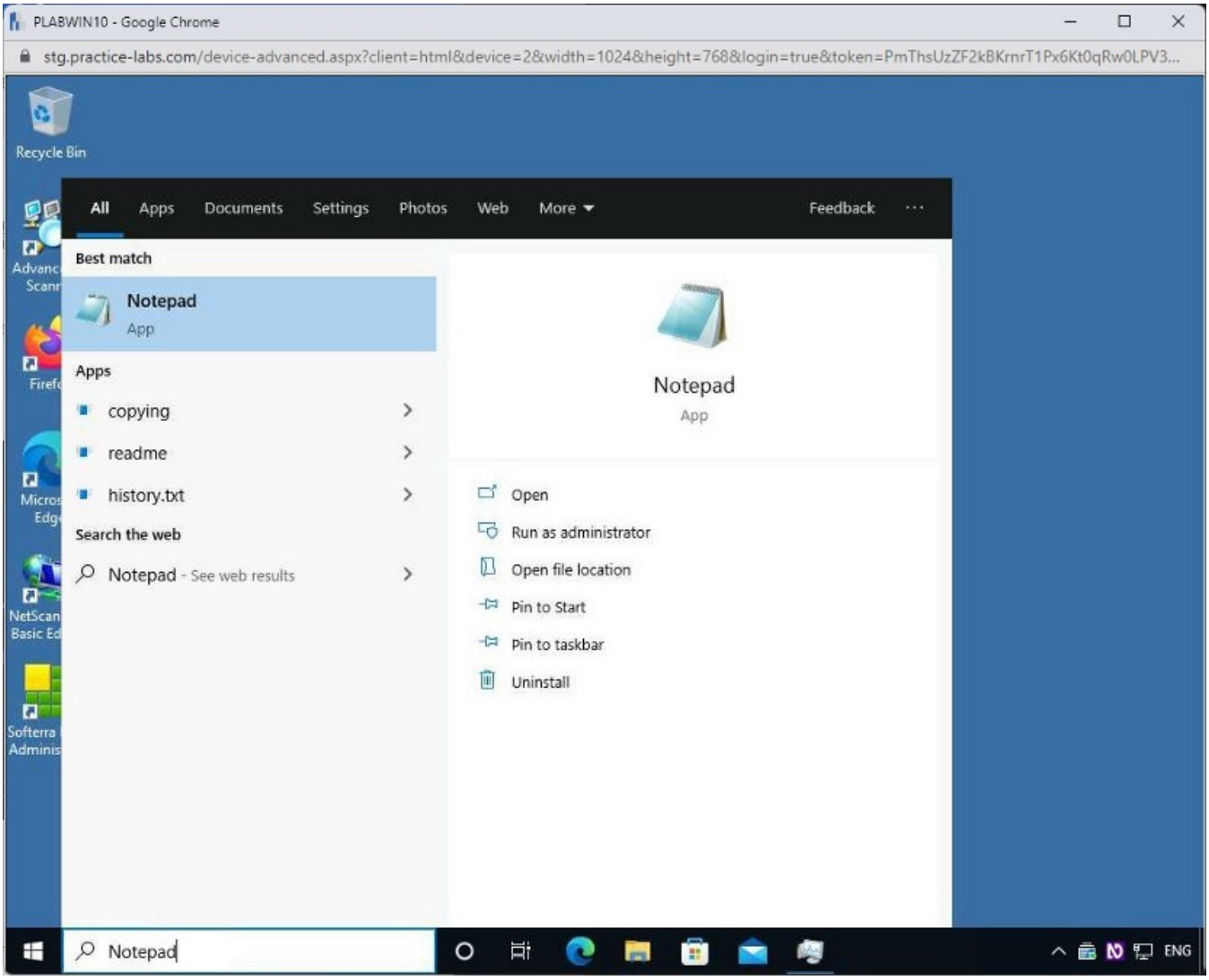
Step 6

You now need to open **Notepad**. You need this to write the virus and then save the file as a batch file. In a real scenario, you could use any text editor available on the system.

In the **Type here to search** textbox on the toolbar, type the following:

```
Notepad
```

From the search results, click **Notepad**.



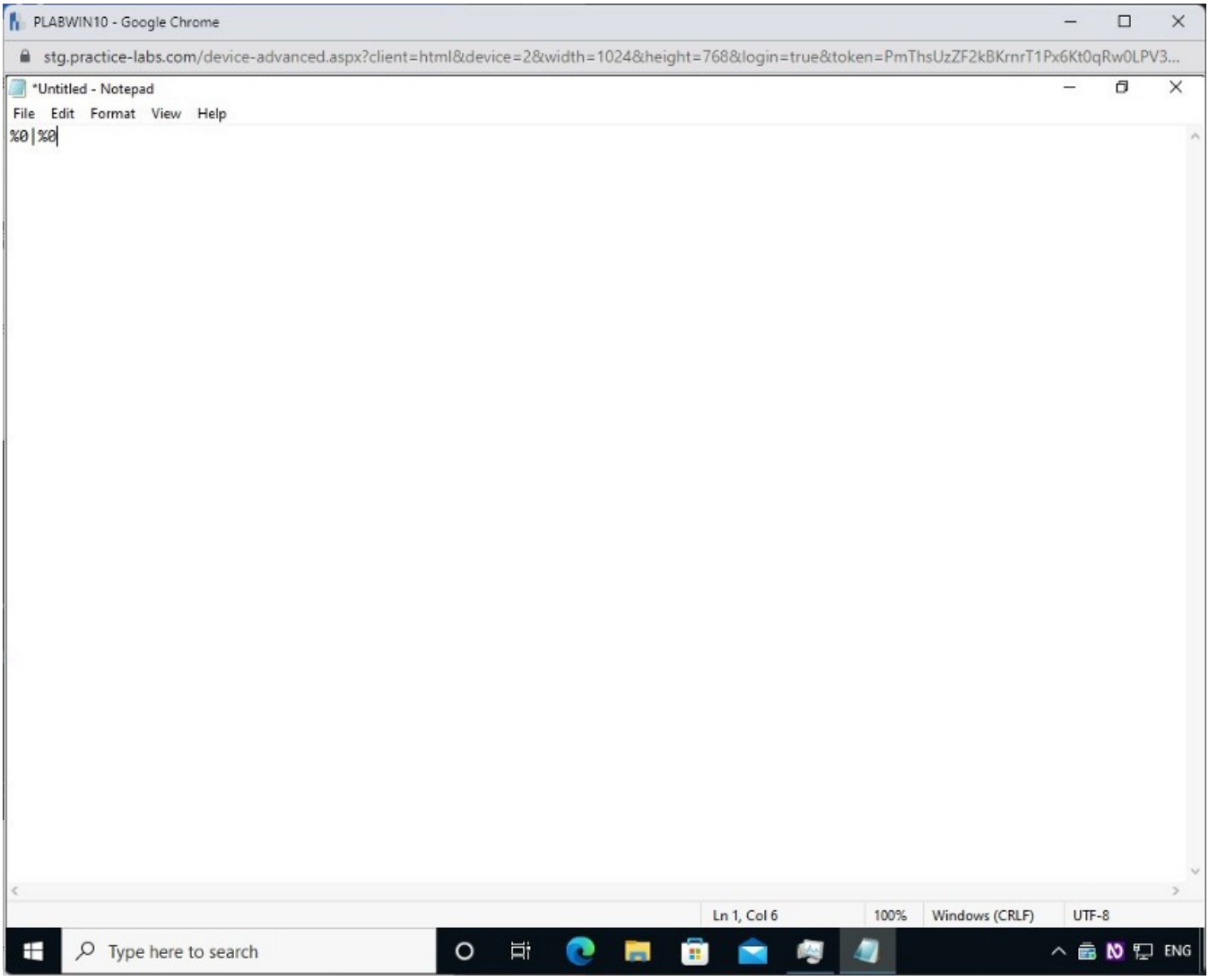
Step 7

The **Untitled — Notepad** window opens.

To create a new batch file, in the **Untitled — Notepad** window, type the following fork bomb code:

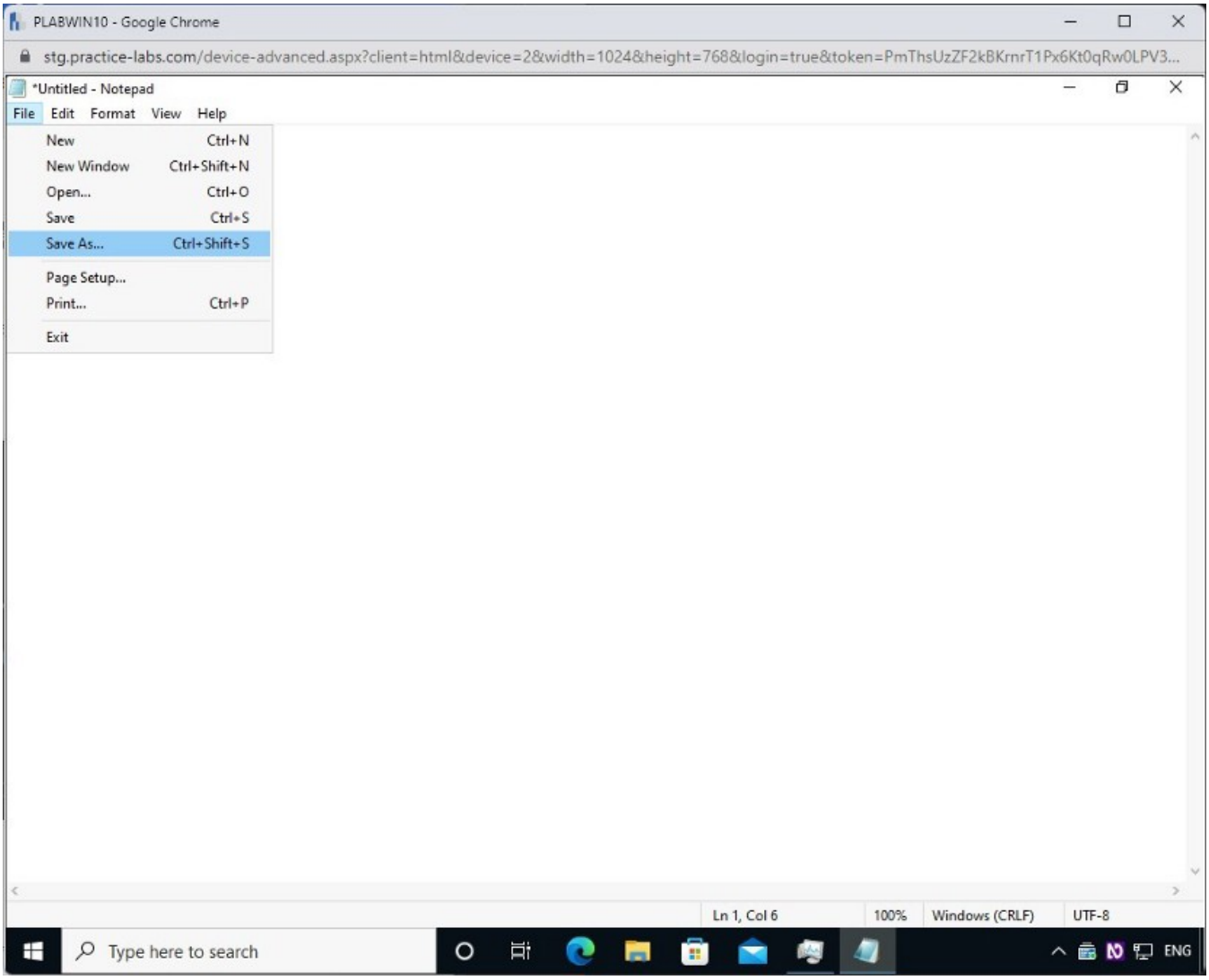
```
%0|%0
```

Note: *A batch file contains instructions to be executed in sequence. In this batch file,* ***%0*** *is the name of the currently executing code. This batch file is going to execute itself forever repeatedly. It quickly creates many processes and slows down the system. First, the* ***%0*** *command is run, and then the second* ***%0*** *command, which is located after the pipe, is run. They both repeatedly run until manually stopped.*



Step 8

To save the file, in the toolbar, click **File** and then click **Save As**.



Step 9

The **Save As** dialog box appears. You can save the file on the desktop. To do this, select **Desktop** in the left pane.

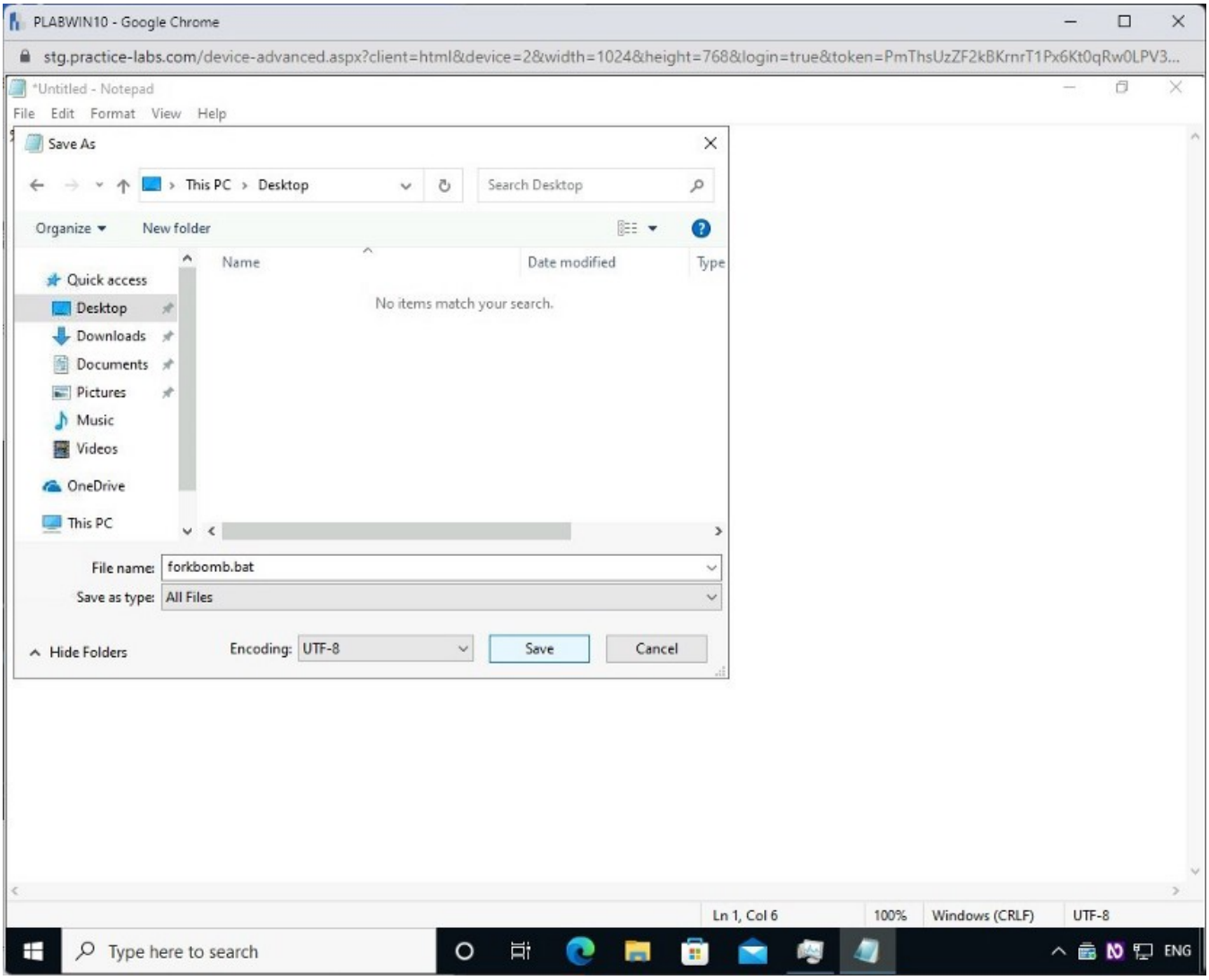
To provide the file name, in the **File name** textbox, type the following:

```
forkbomb.bat
```

From the **Save as type** drop-down, select **All Files**.

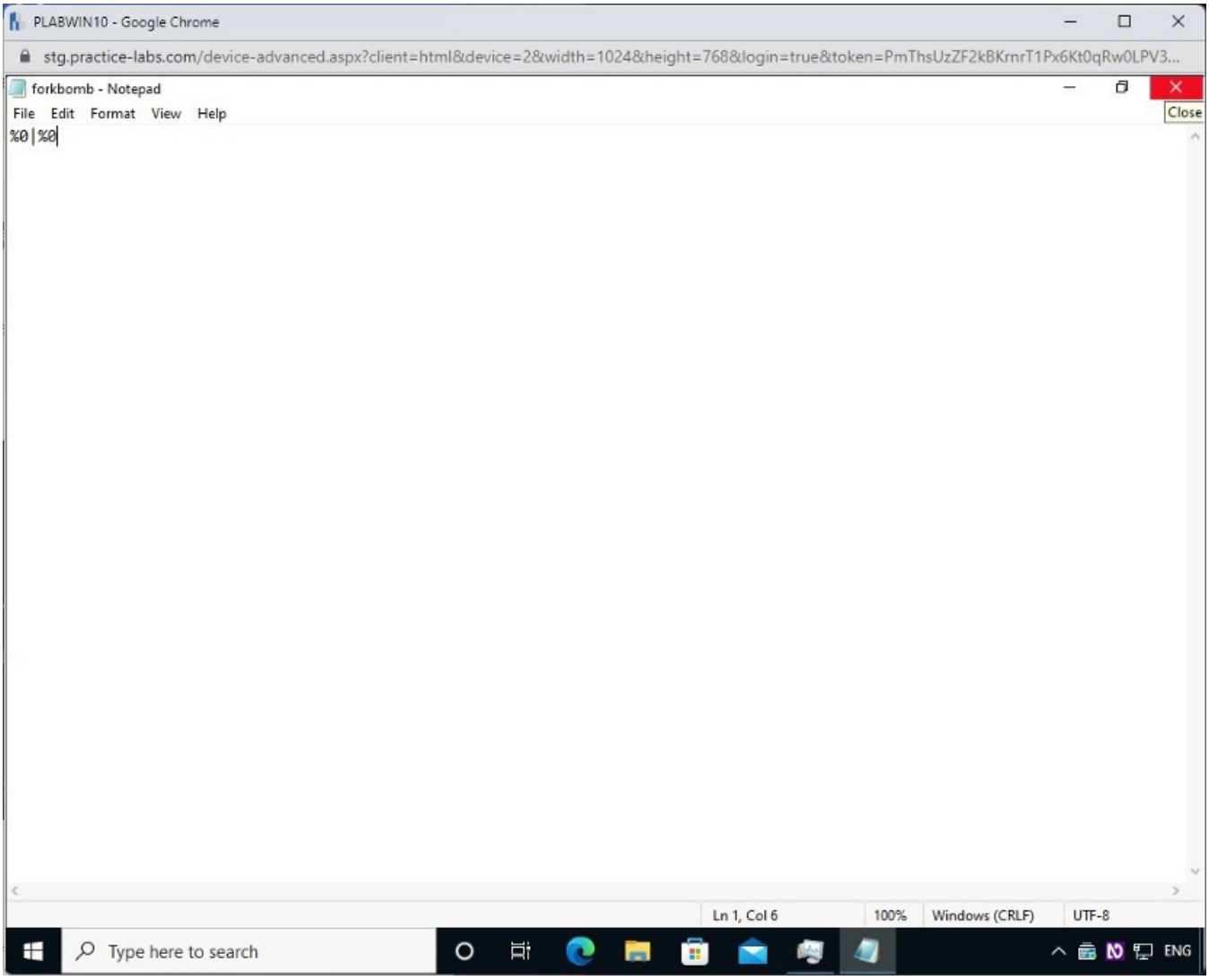
Alert: If you do not choose **All Files**, the file will be saved with the default **.txt** extension. With the selection of **All Files**, you can provide any other extension with the file name.

Click **Save**.



Step 10

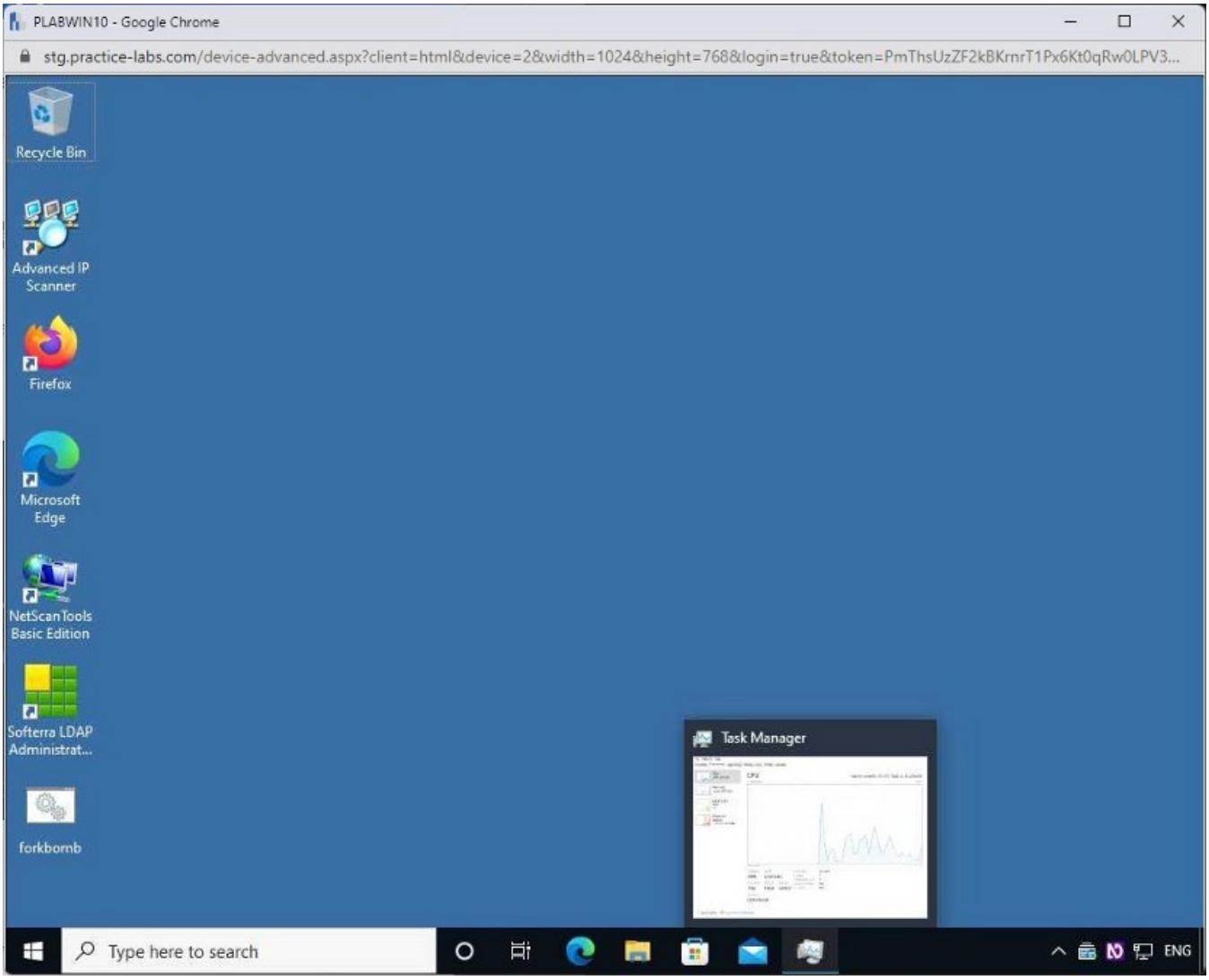
Close the **forkbomb — Notepad** window.



Step 11

Notice that the **forkbomb.bat** file is created on the desktop.

You need to restore **Task Manager** now. Click **Task Manager** in the taskbar.



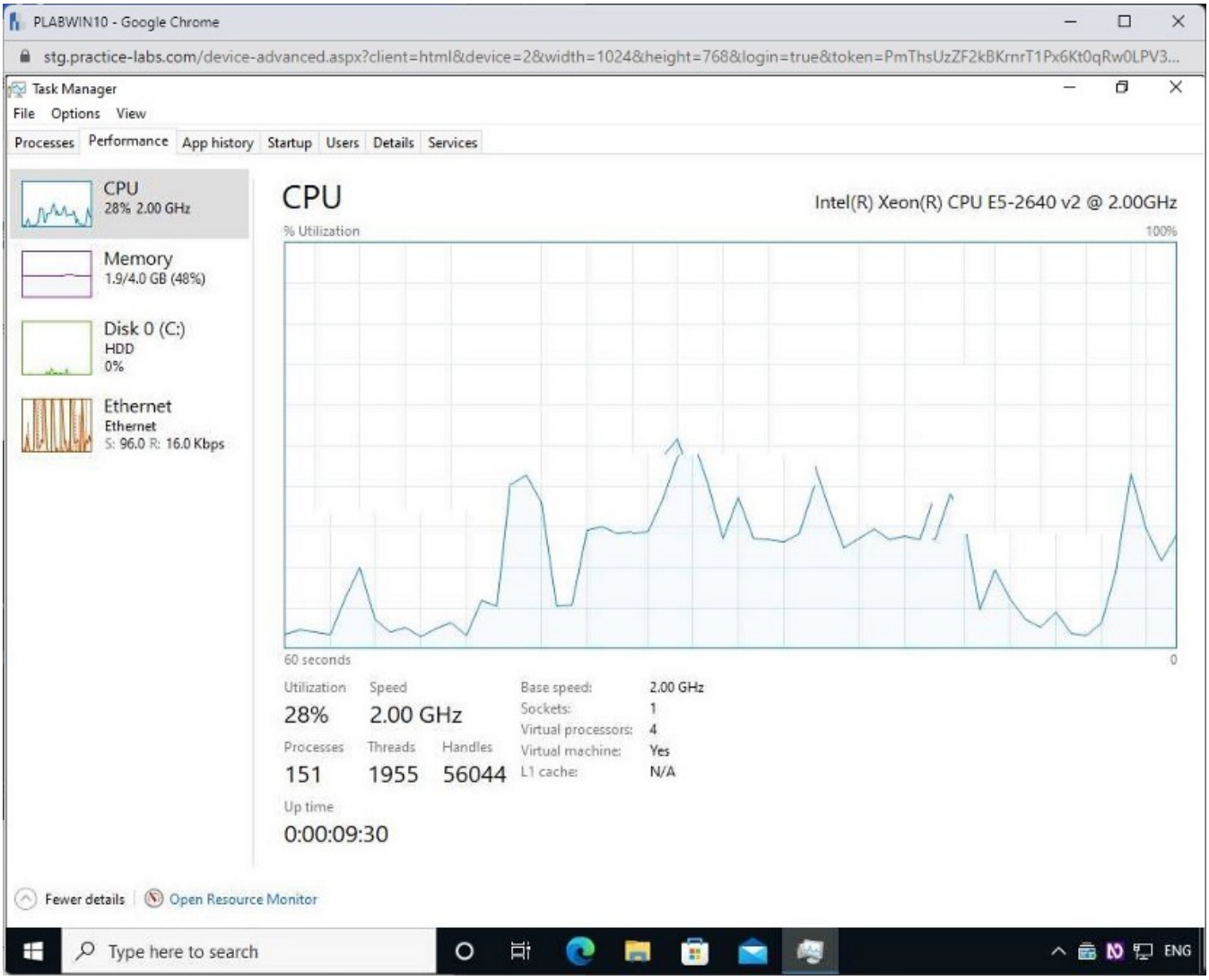
Step 12

Before you execute the **forkbomb.bat** batch file, observe the **CPU** usage in **Task Manager**.

Under the Performance tab, observe the CPU activity in the left-hand pane.

The **CPU** utilization is quite low.

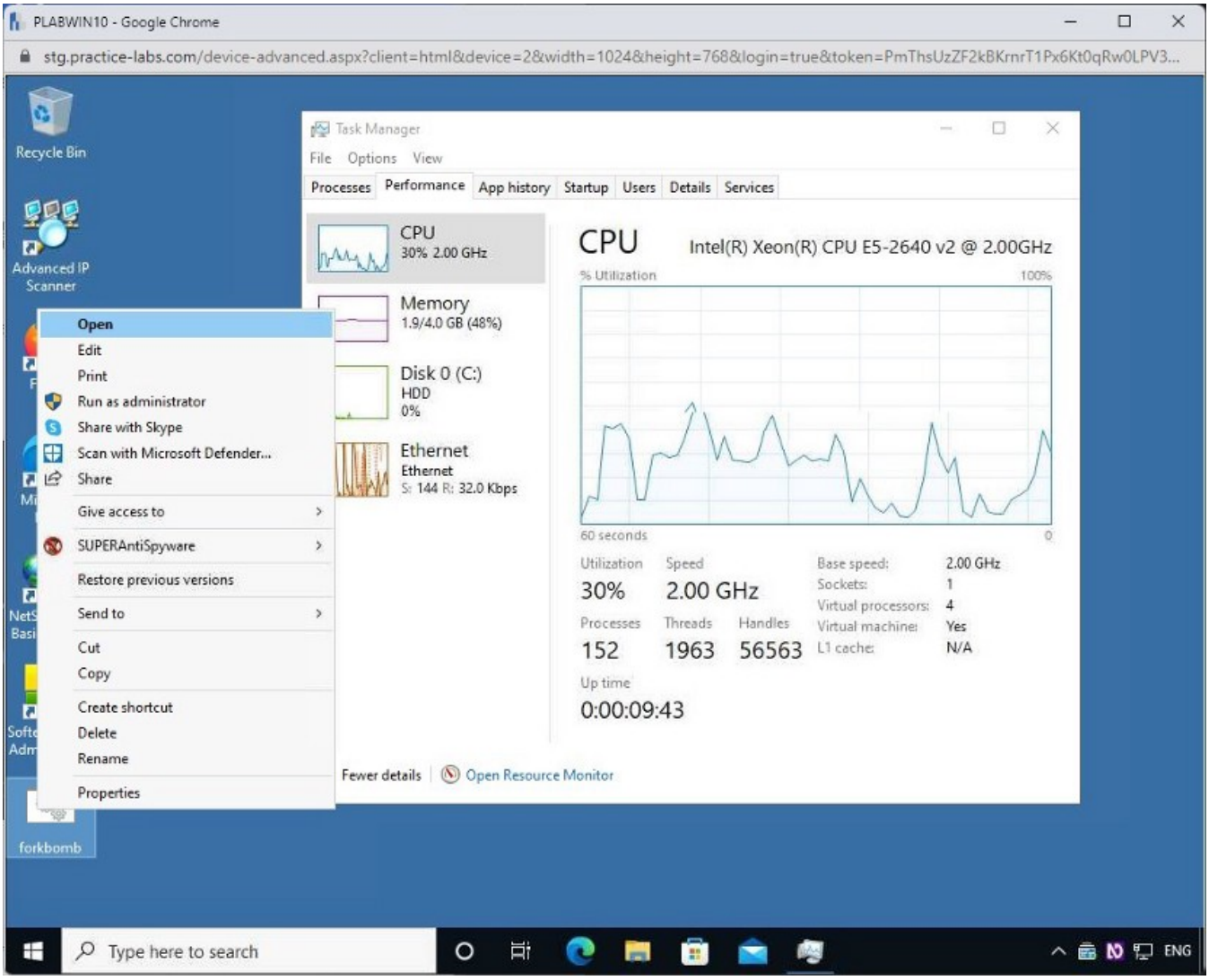
Note: *The CPU performance will vary in your lab environment.*



Step 13

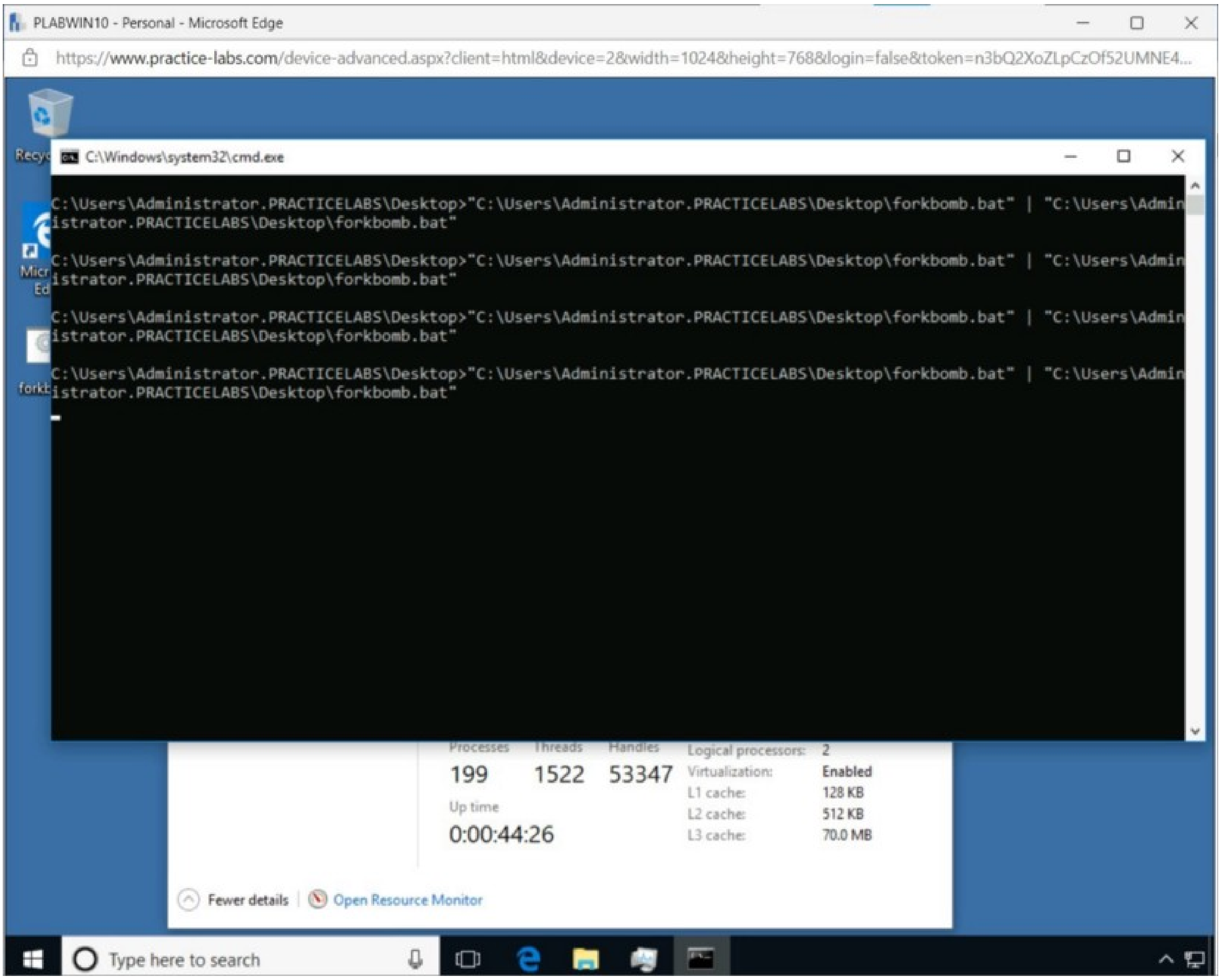
Reduce the size of the **Task Manager** window so that the **forkbomb.bat** file is visible on the desktop.

To execute the **forkbomb.bat** file on the desktop, right-click **forkbomb** and select **Open**.



Step 14

The **Command Prompt** window opens, and the **forkbomb.bat** file starts executing recursively.



Step 15

After the batch file execution, observe the CPU usage in **Task Manager**.

Observe the CPU activity in the left-hand pane under the **Performance** tab in the **Task Manager** window, which may be hidden under the dialog boxes.

You may receive an error message during the execution.

1

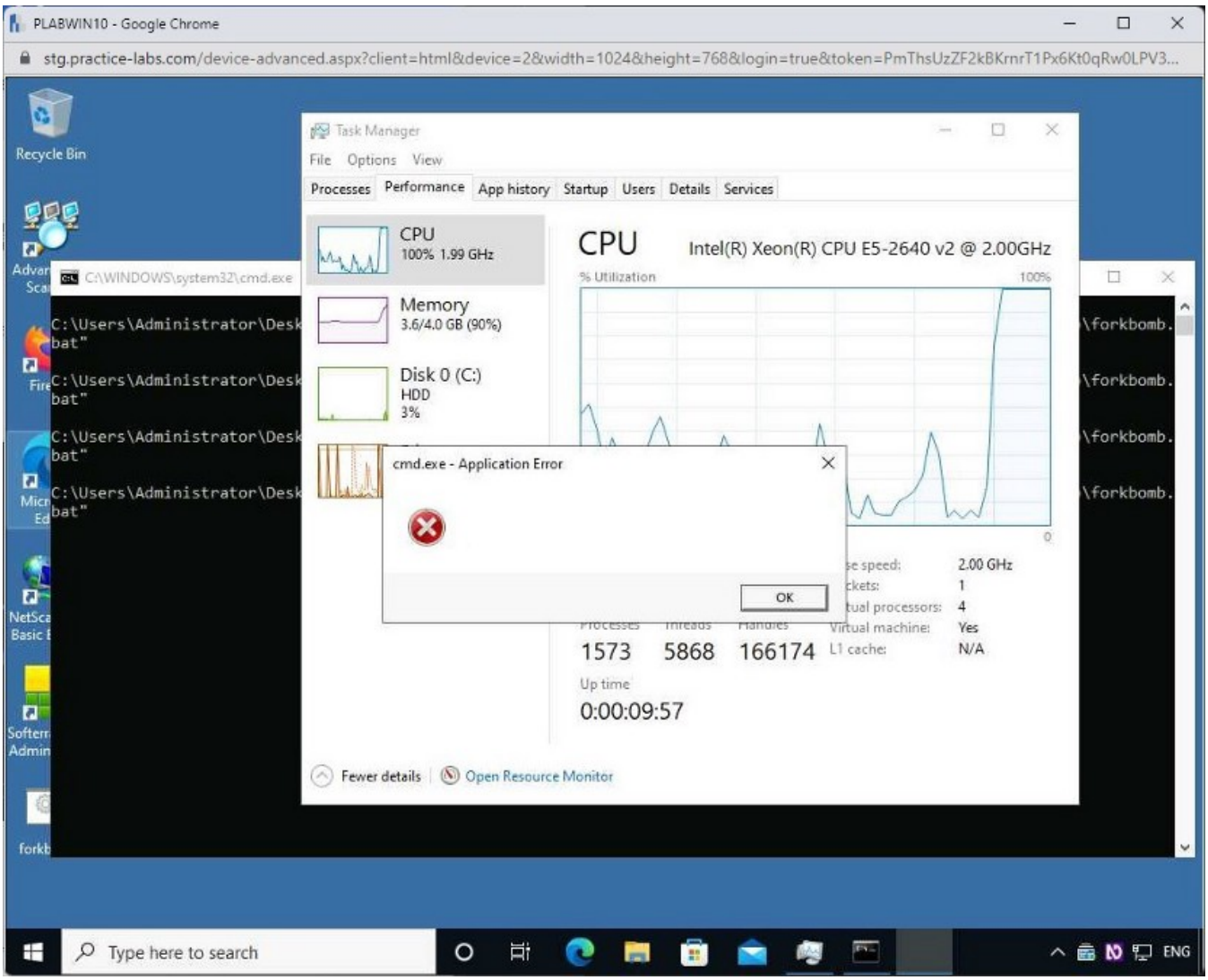
Screenshot

Click the button to take a screenshot of PLABWIN10

Take screenshot

1 of 8

Note: *The CPU and memory usage may vary in your lab environment. You may not even bring the* ***Task Manager*** *in front and see that the* ***Windows Command Processor*** *has stopped the message.***Alert**: The **PLABWIN10** device may hang and could crash. Windows may abruptly close, and you may not be able to reconnect to the device. To stop the execution of the virus, you can **Reboot** the device in the central pane of the Practice Labs platform. After this, you will be reconnected and continue to the next exercise.



Keep all windows open and proceed to the next exercise.

Exercise 2 — Advanced Persistent Threat

An Advanced Persistent Threat (APT) is a type of network attack is which an attacker gains unauthorized access to a network without permission and remains undetected for a long period of time. APTs are often associated with Nation States and see to obtain sensitive information rather than taking systems down.

In this exercise, you will learn about APTs, it's attributes, and it's lifecycle.

Learning Outcomes

After completing this exercise, you will have further knowledge of:

- Advanced Persistent Threat (APT)
- Attributes of APTs
- Lifecycle of APTs

Advanced Persistent Threat (APT)

An APT is a long-term network attack performed by highly skilled and well-funded hackers who seek to gain access to sensitive information without being detected. To this end, most APTs do not seek to destroy assets nor cause a denial of service. Often these attacks last for many months and even into years.

To understand an APT, let's look at each word:

- **Advanced**: The attack is conducted with highly skilled hackers and sophisticated tools.
- **Persistent**: The attack is perpetrated over a long period of time. The expertise of the hackers, and the level of funding the hackers receive allow them to use highly sophisticated tools to create a stealth command and control structure within the target network.
- **Threat**: The hacking team is generally well-funded, highly-organized, and, most often, performed by government bodies going after defense, aerospace, financial or other highly valuable information.

Lifecycle of APTs

Like any attack, APTs follow a specific lifecycle. Using the steps in this lifecycle, they can penetrate through the traditional security controls to gain access to the networks. To better understand the lifecycle of APTs, let's look at each phase.

PreparationIntrusionExpansionPersistenceSearch andExfiltrationCleanupAPT

Figure 2.1: Lifecycle of an APT: Phases in the APT lifecycle

PreparationIn the first phase, The APT team prepare themselves well. They search for the target, study it, and put a team together.**Intrusion**After the preparation phase,

hackers gear up for the initial intrusion which they use various methods to initiate, such as can use social engineering, drive-by-download, or even spear-phishing to target personnel within an organization. Using one of these methods, malware is deployed on the network.**Expansion**In this phase, hackers irst obtain a user's credentials and then perform privilege escalation to hopefully access administrative credentials. Finally, the team can pivot or perform lateral movement to access other areas of the network.**Persistence**After the team has established themselves within a network, they must remain in the system undetected to achieve their final objective. Hackers will utilize a series of customized tools and malware that will be undetected by anti-virus or other security tools. Additionally, they can place the malware in locations that are not commonly used like routers, servers, registries, firewalls, or even printers.**Search and Exfiltration**

Knowing where the sensitive data is within a system is difficult to ascertain. Often hackers will simply steal all data they can locate and look for important data once one their own network, or they can perform searches or scans inside the target network to find and only exfiltrate that data. Either way, they must evade data loss prevention (DLP) tools when exfiltrating. Encryption or placing small amounts of data in packet headers are two of the methods the hacking team can use to accomplish this.

Cleanup

It is of utmost importance that an attacker restores systems before leaving to remove any trace they were ever there. This can be accomplished by deleting log files, changing file attributes to the original, or even manipulating data to mislead security professionals.

Exercise 3 — Antimalware Programs

Most operating systems have built-in anti-malware programs that may help malware removal, unwanted email messages, spyware, and other annoyances that can cause anomalies within a computer system.

In this exercise, you will explore a built-in anti-malware application in Windows 10 called **Windows Defender**. You will also install anti-spyware and use it to scan **PLABWIN10** as examples of different tools used to detect and remove Malware.

Learning Outcomes

After completing this exercise, you will be able to:

- Scan Using Windows Defender
- Using an Online Anti-Malware Scanner
- Use SUPERAntiSpyware

Task 1 — Scan Using Windows Security

Windows Security is the built-in anti-malware software in Windows 10. You must keep this software updated with the latest signatures to keep malware at bay while using your computer.

To update **Windows Security**, follow these steps:

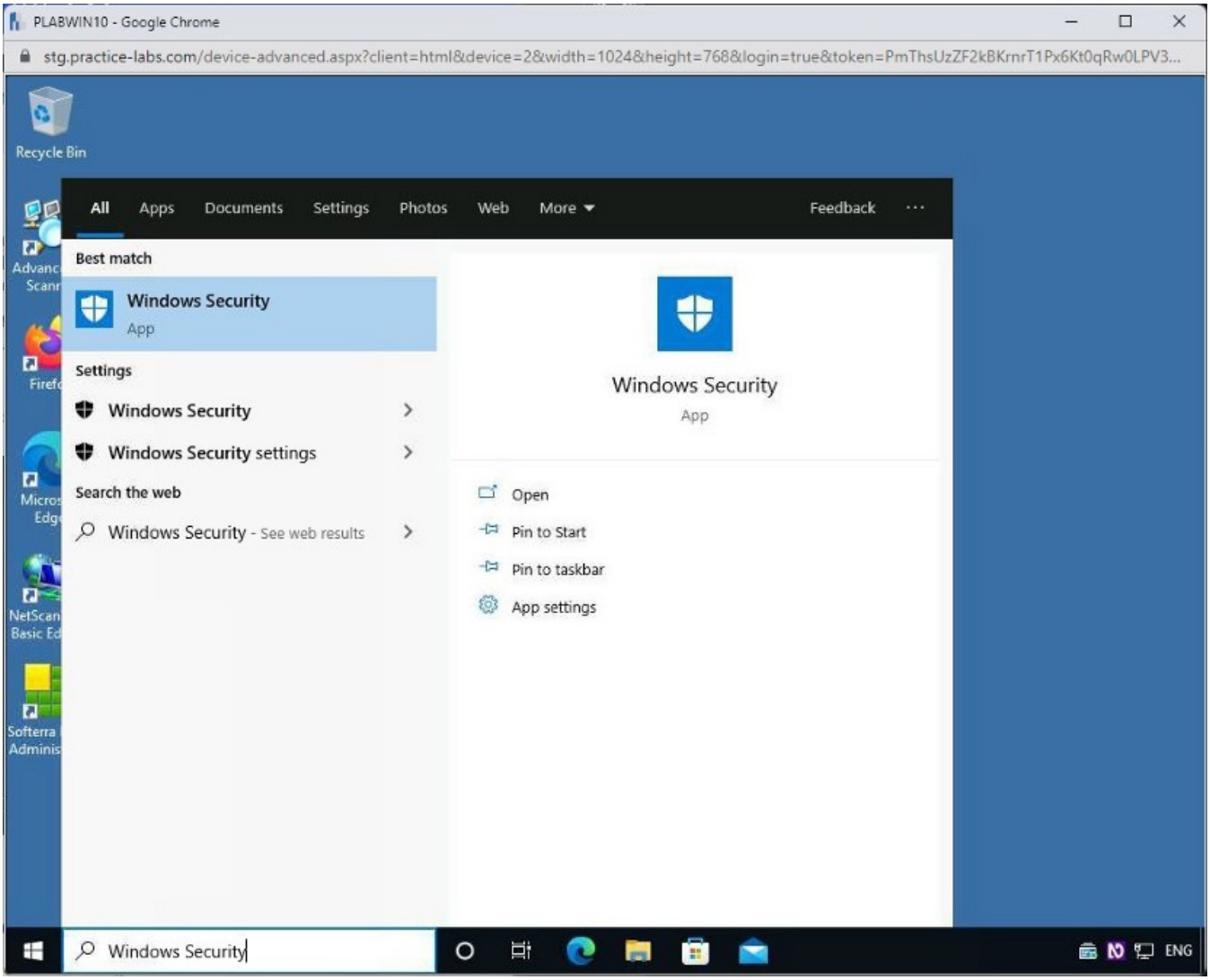
Step 1

Ensure you have powered on the required devices and connect to **PLABWIN10**.

In the **Type here to search** textbox, type the following:

```
Windows Security
```

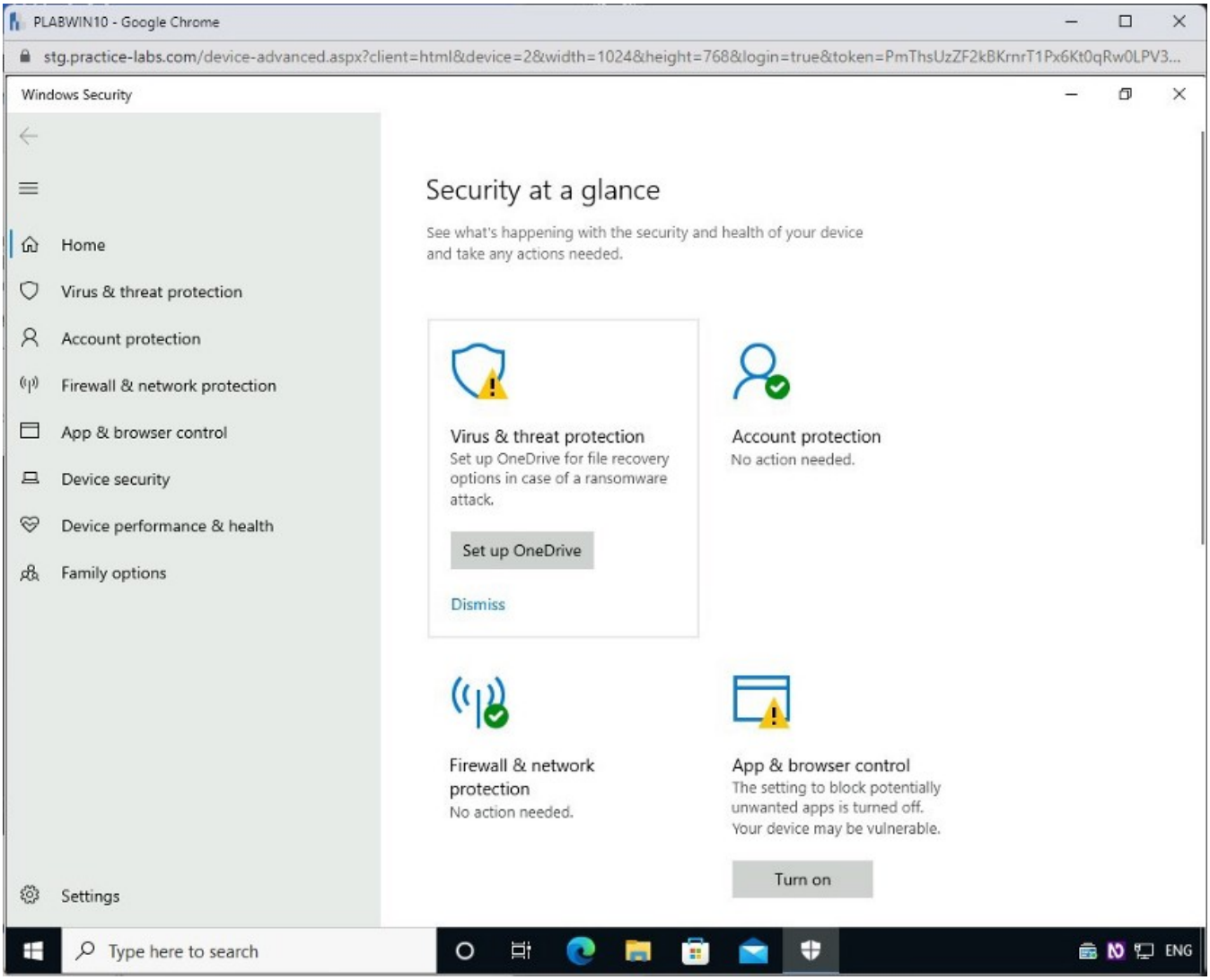
From the **Search** results, click **Windows Security**.



Step 2

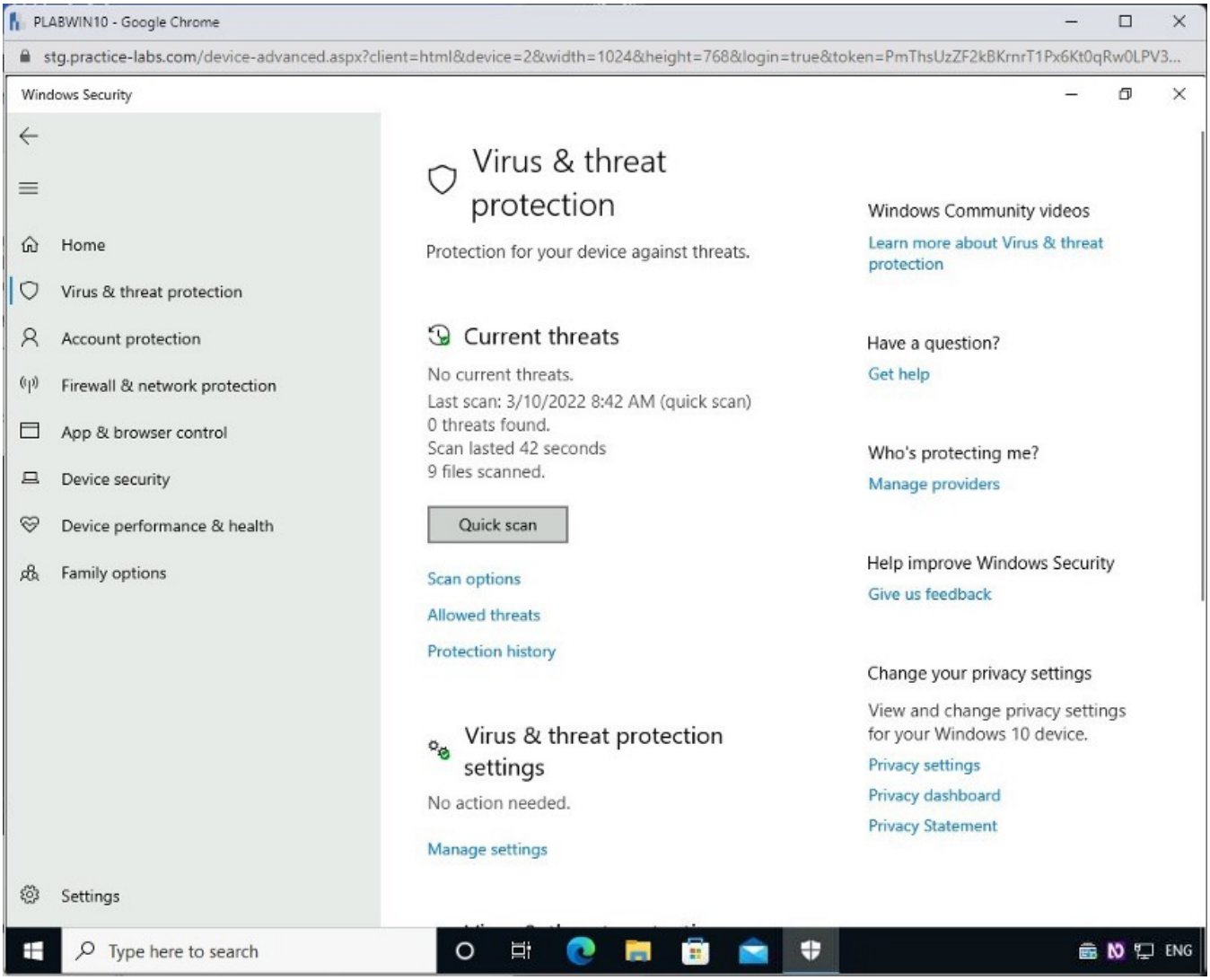
The **Windows Security** window is displayed.

Click **Virus & threat protection**.



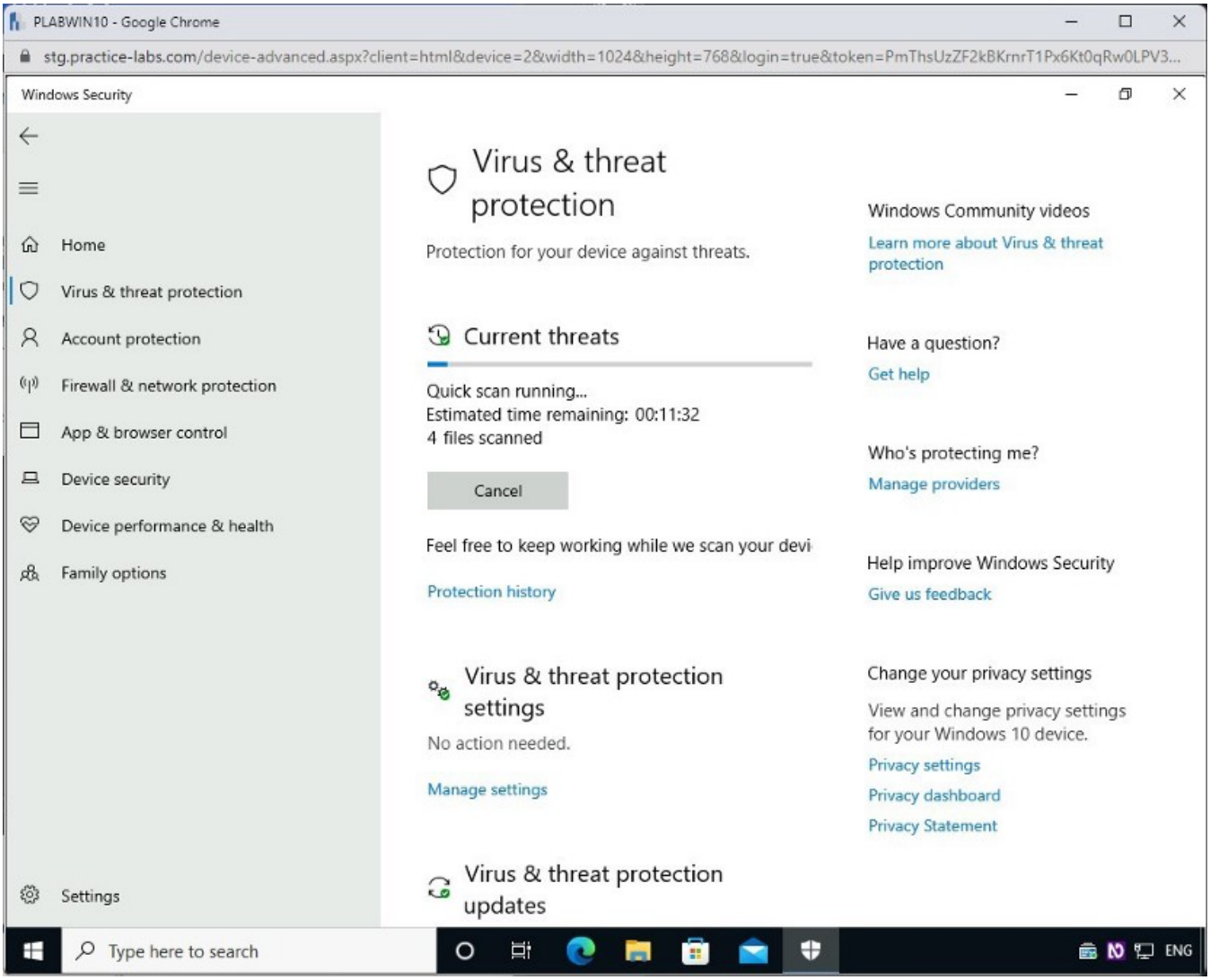
Step 3

On the **Virus & threat protection settings** page, click **Quick scan**.



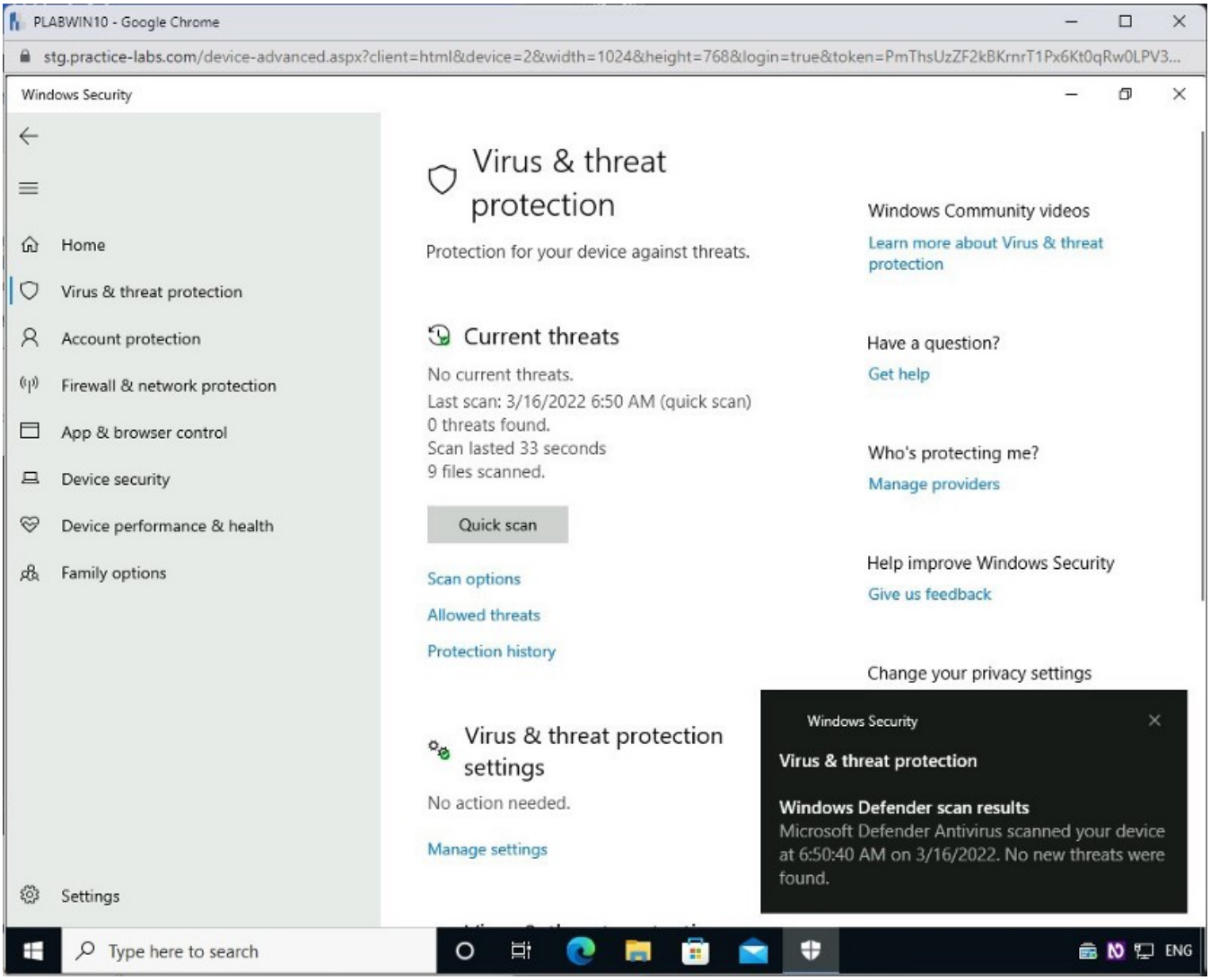
Step 4

The scan is in progress.



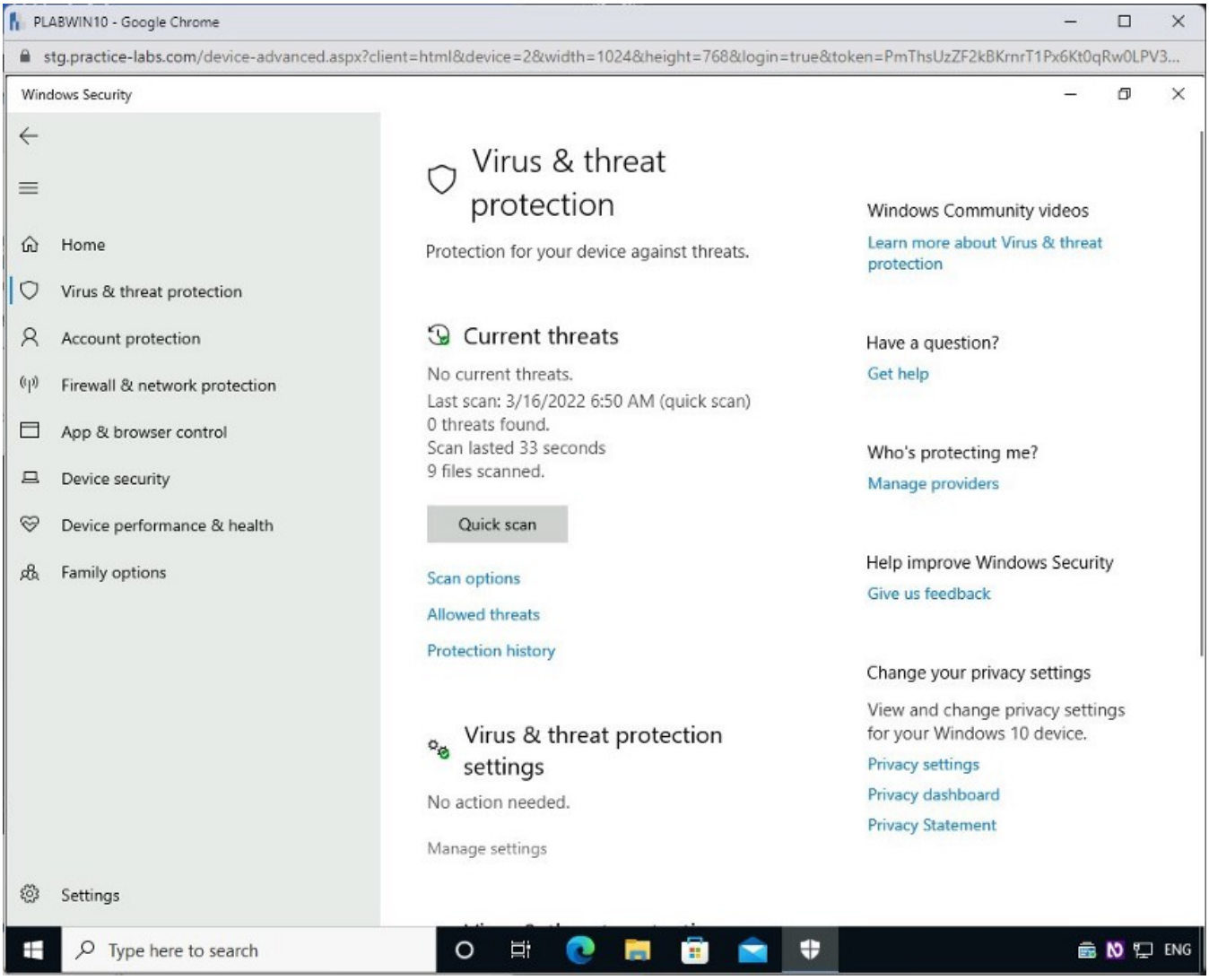
Step 5

A notification at the bottom right corner displays the scan results.



Step 6

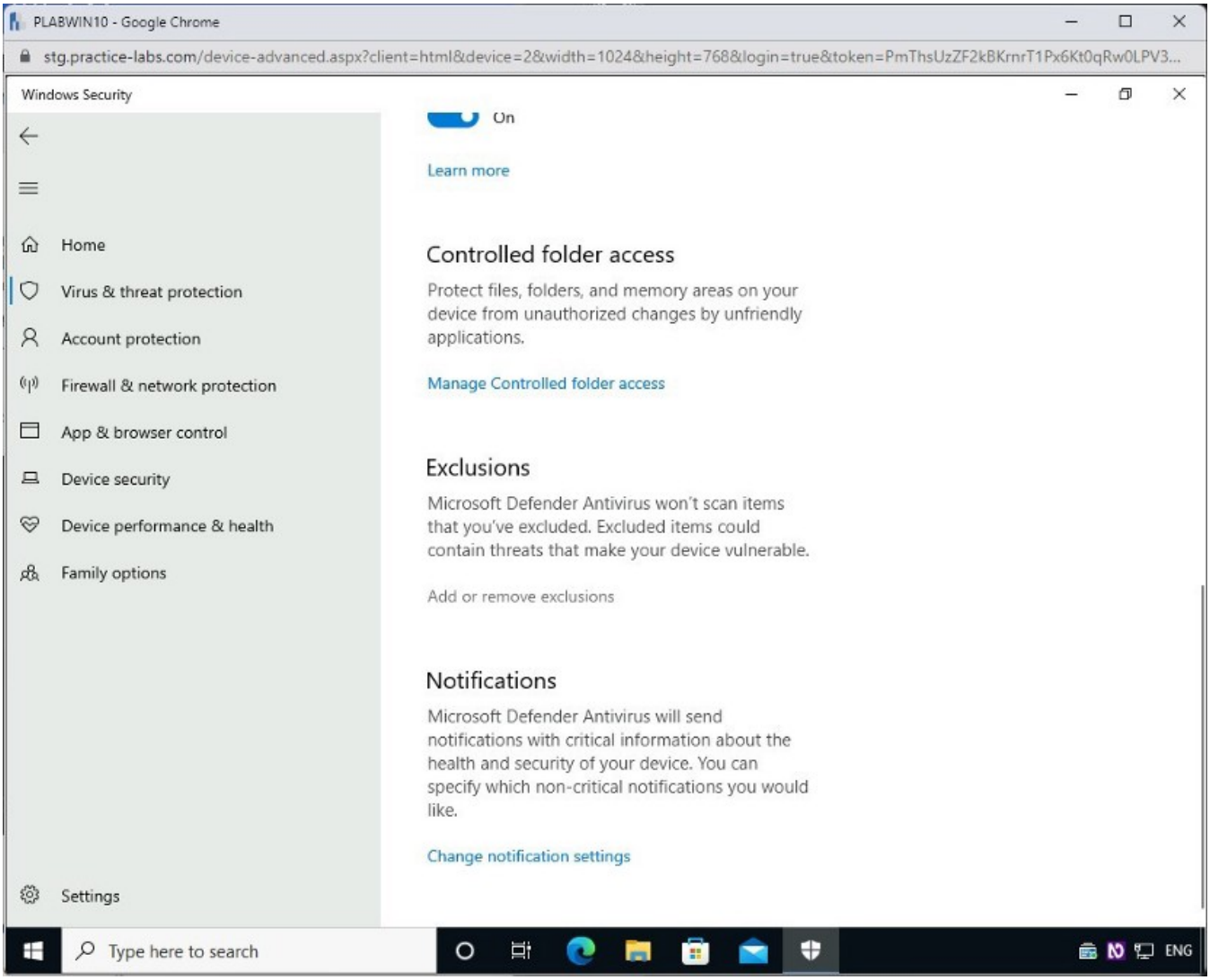
On the **Virus & threat protection** page, click **Manage Settings** under **Virus & threat protection settings**.



Step 7

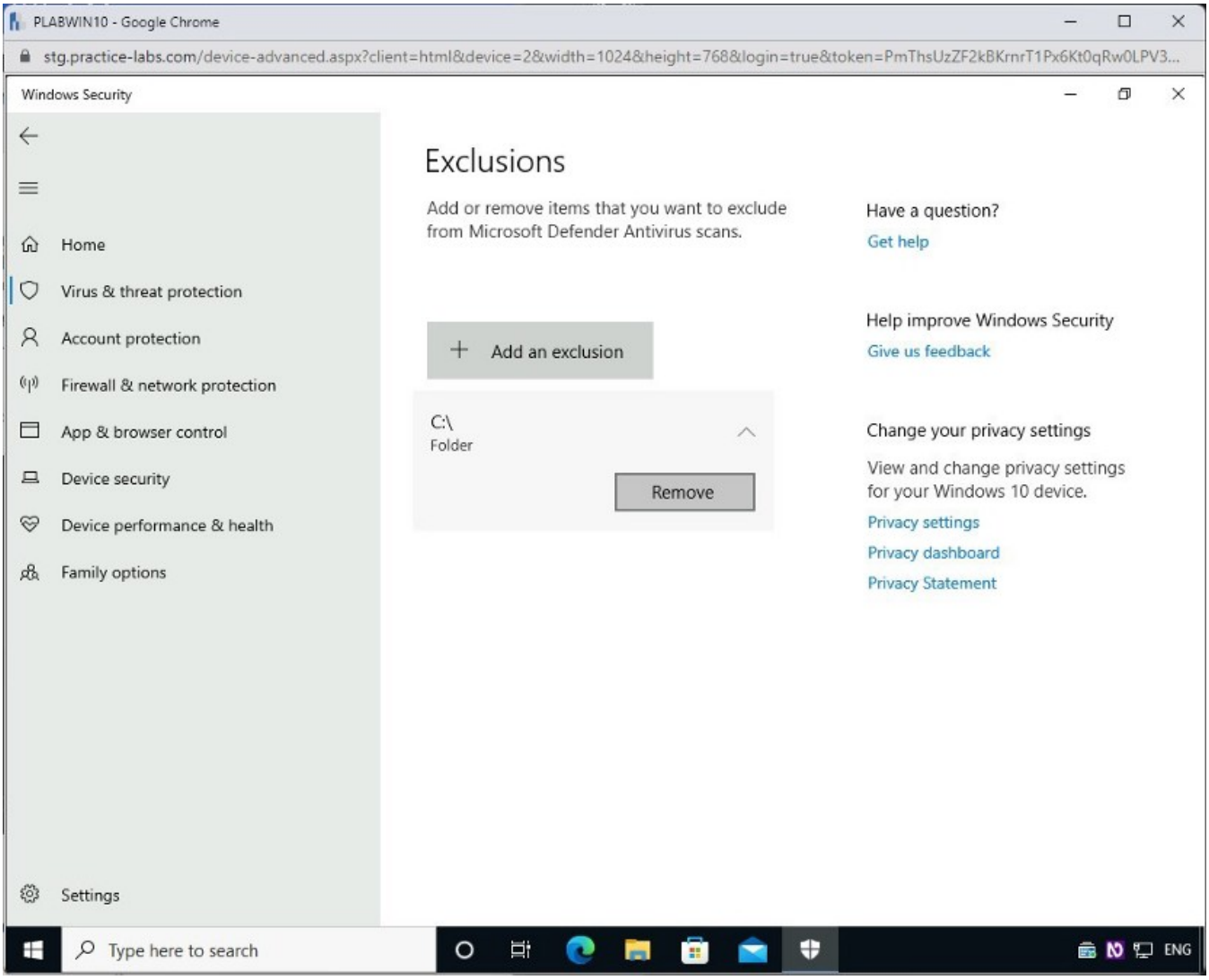
Scroll to the **Exclusions** section on the **Virus & threat protection settings** page.

Click **Add or remove exclusions**.



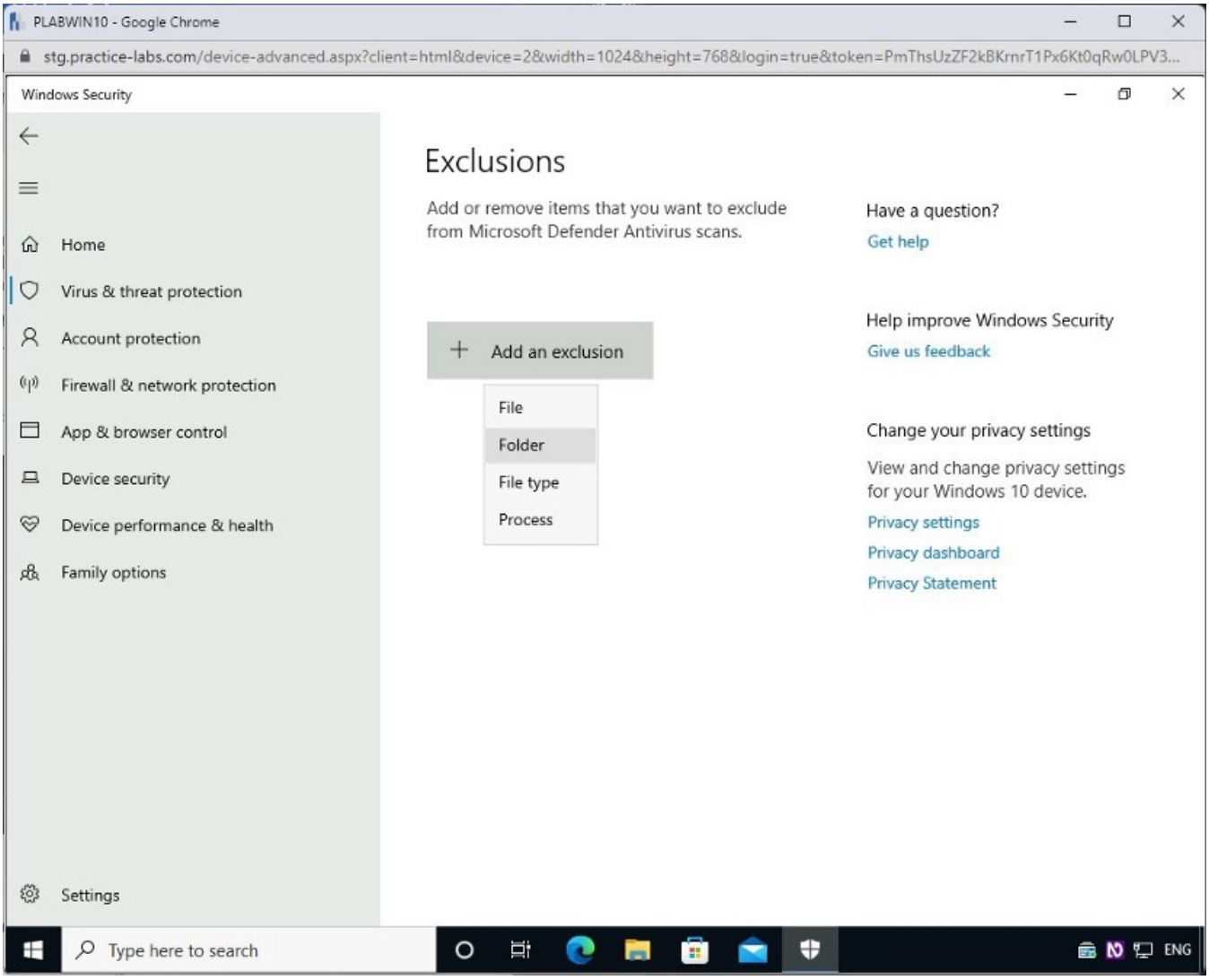
Step 8

On the **Exclusions** page, click the down arrow after **C:** and click **Remove**.



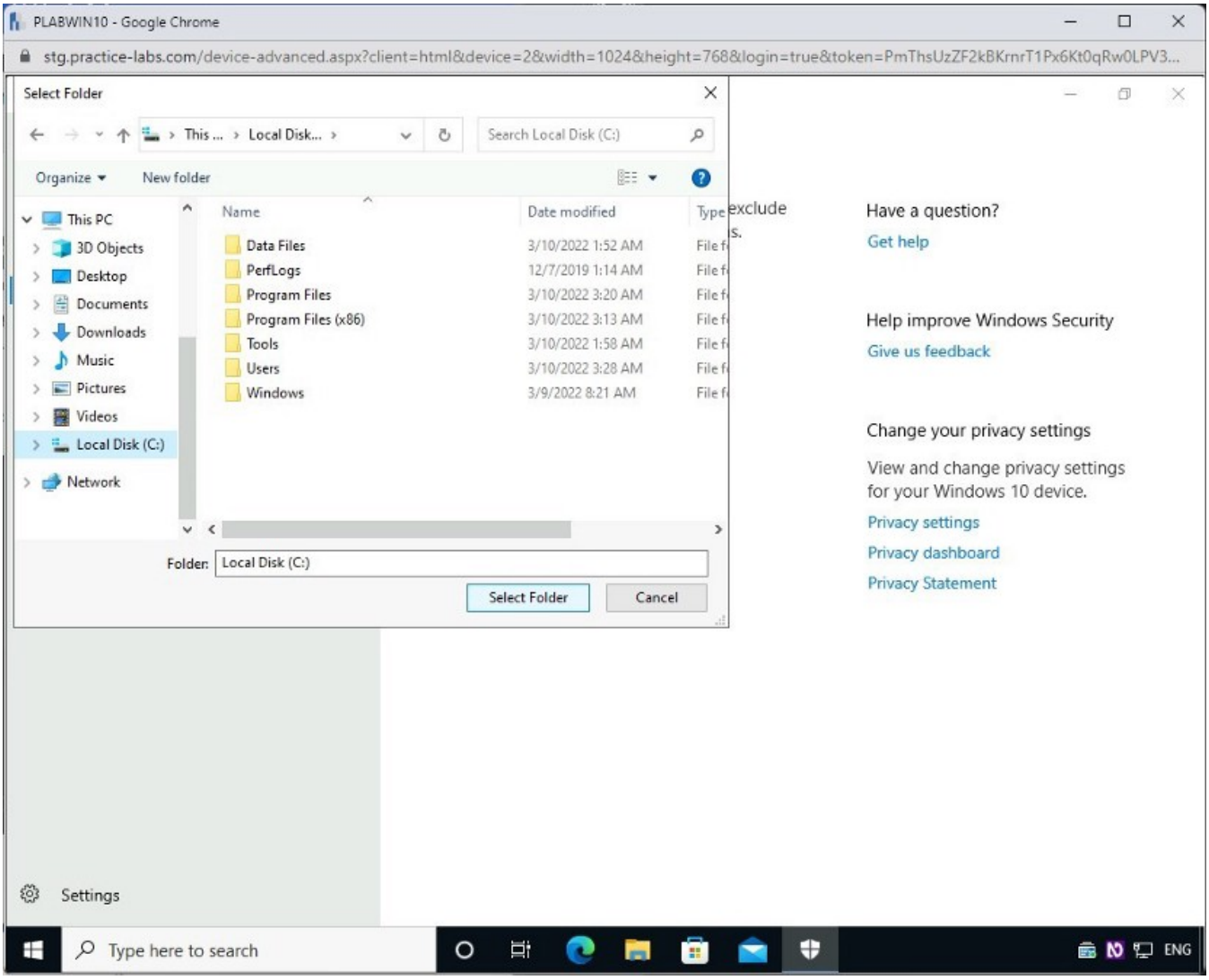
Step 9

Click the + **(Add an exclusion)** sign and then select **Folder**.



Step 10

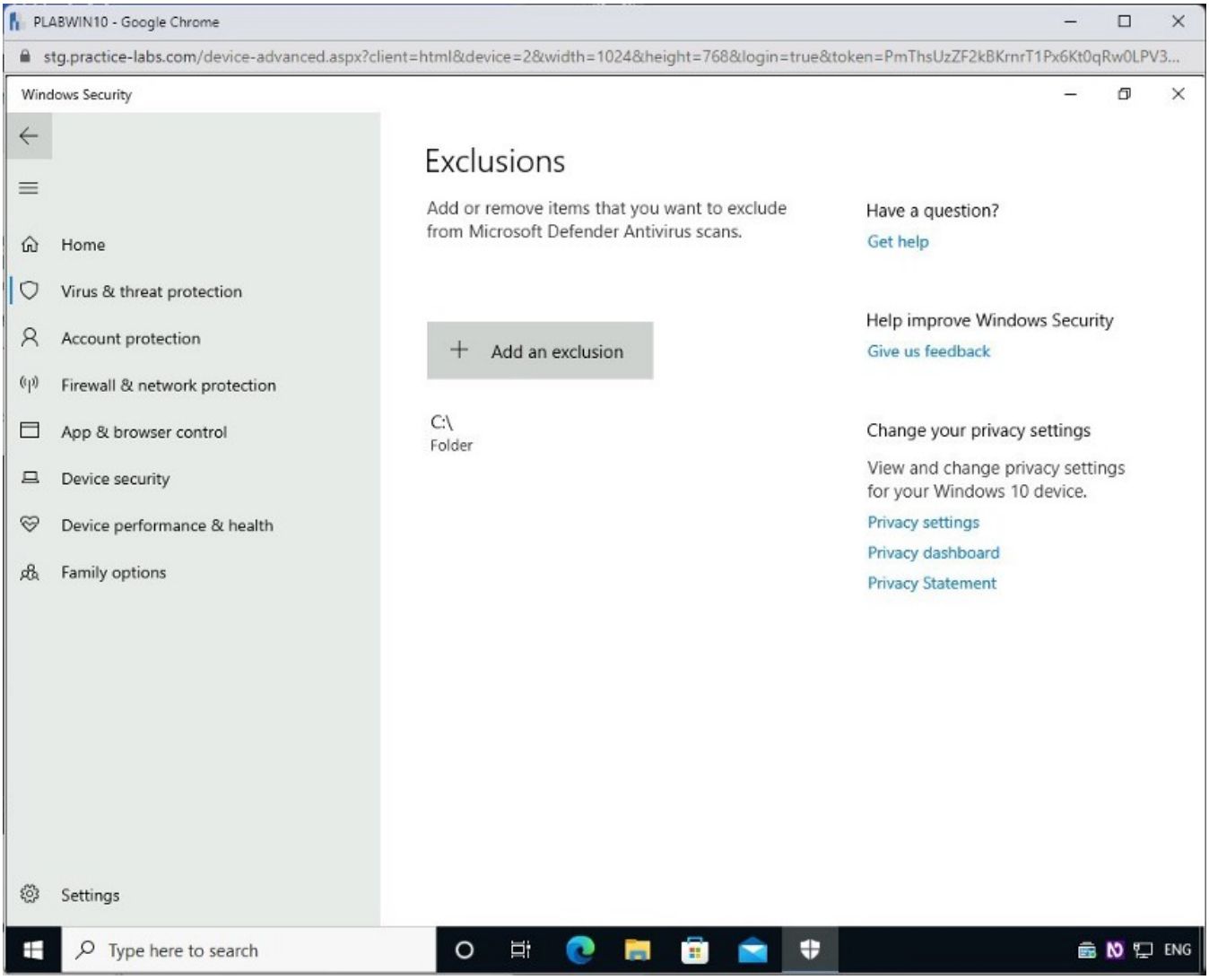
The **Select Folder** dialog box is displayed. Navigate to the **C Drive**, select **Local Disk (C:)** and then click **Select Folder**.



Step 11

This folder is now added to the exclusions.

***Note**: In this example we have shown the **C:** drive, although you can exclude any files, folders file types or processes using this tool.*



Close the Windows Security window.

Task 2 — Using an Online Anti-Malware Scanner

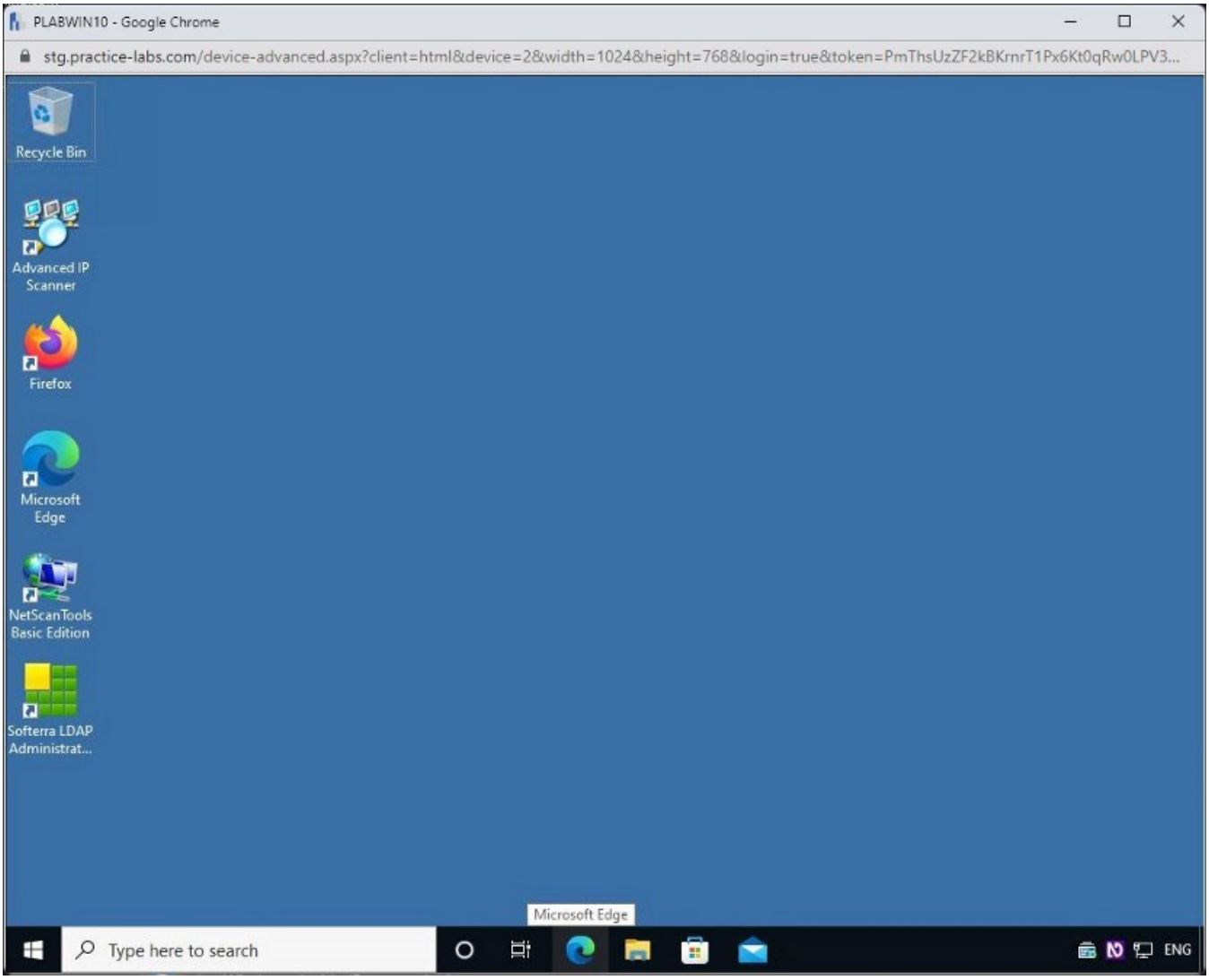
In addition to anti-malware programs installed locally on a computer, you can use free online anti-virus scanners to check your system for suspicious activity.

Please note that web page appearance may change, and URL links may become outdated. You can use your search engine to find online anti-malware scanners.

Step 1

Ensure you are connected to **PLABWIN10**.

Open **Microsoft Edge** by clicking on the icon on the taskbar.



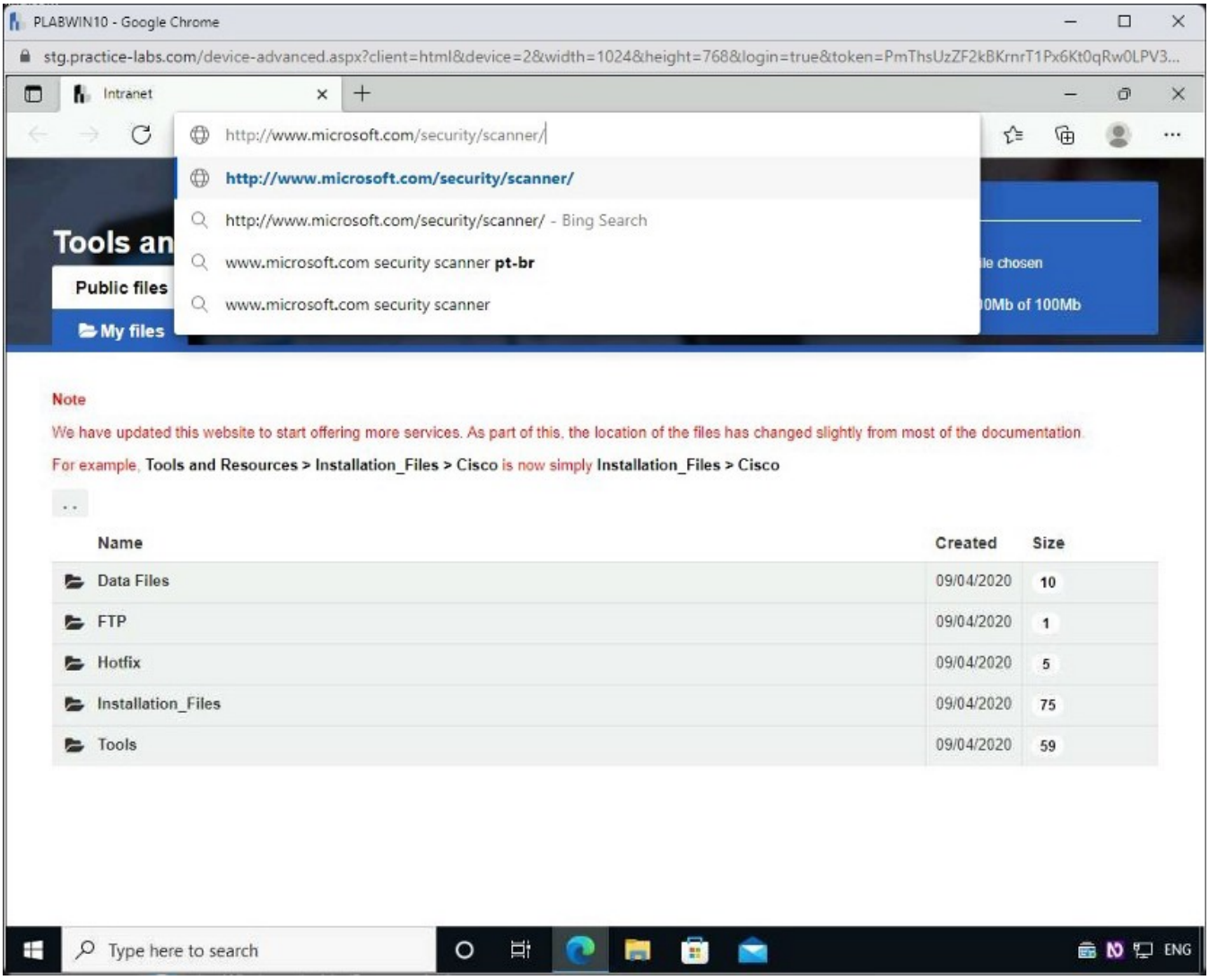
Step 2

In the address bar, type the following URL:

```
http://www.microsoft.com/security/scanner/
```

Press **Enter**.

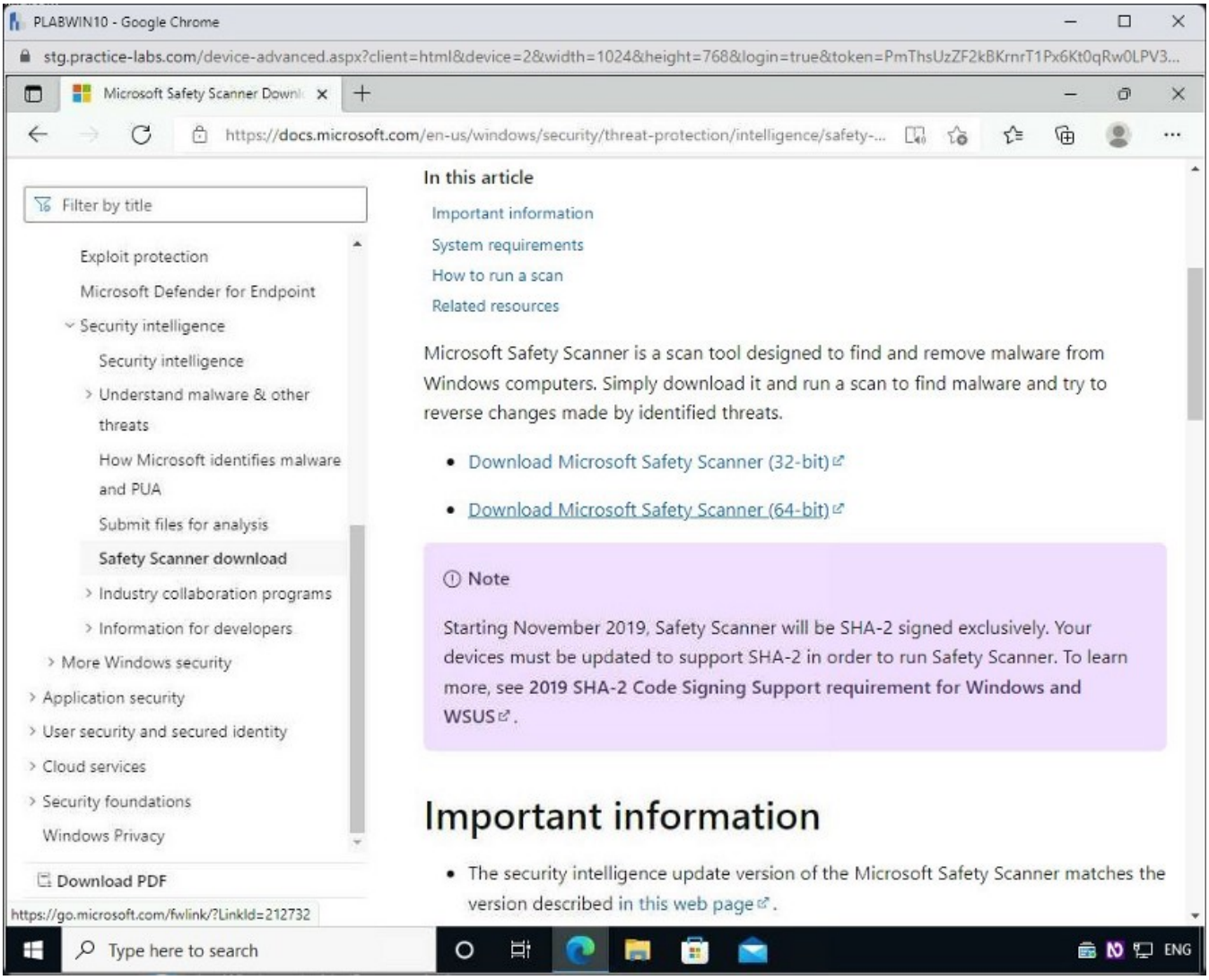
***Note**: If the URL indicated above did not work, use your search engine to search for Microsoft's online safety and security center.*



Step 3

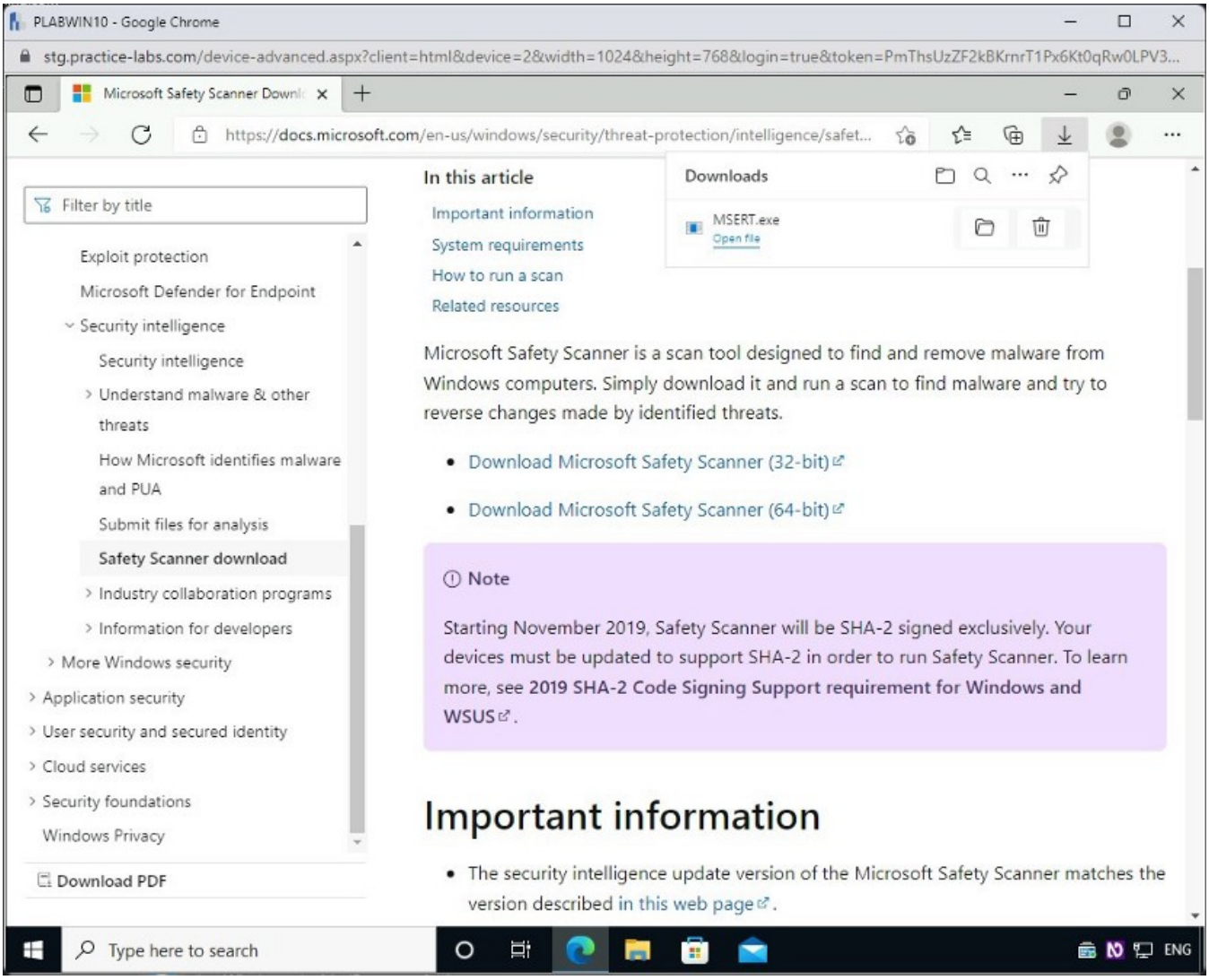
The **Microsoft Safety Scanner** page is displayed.

Click **Download Microsoft Safety Scanner (64-bit)**.



Step 4

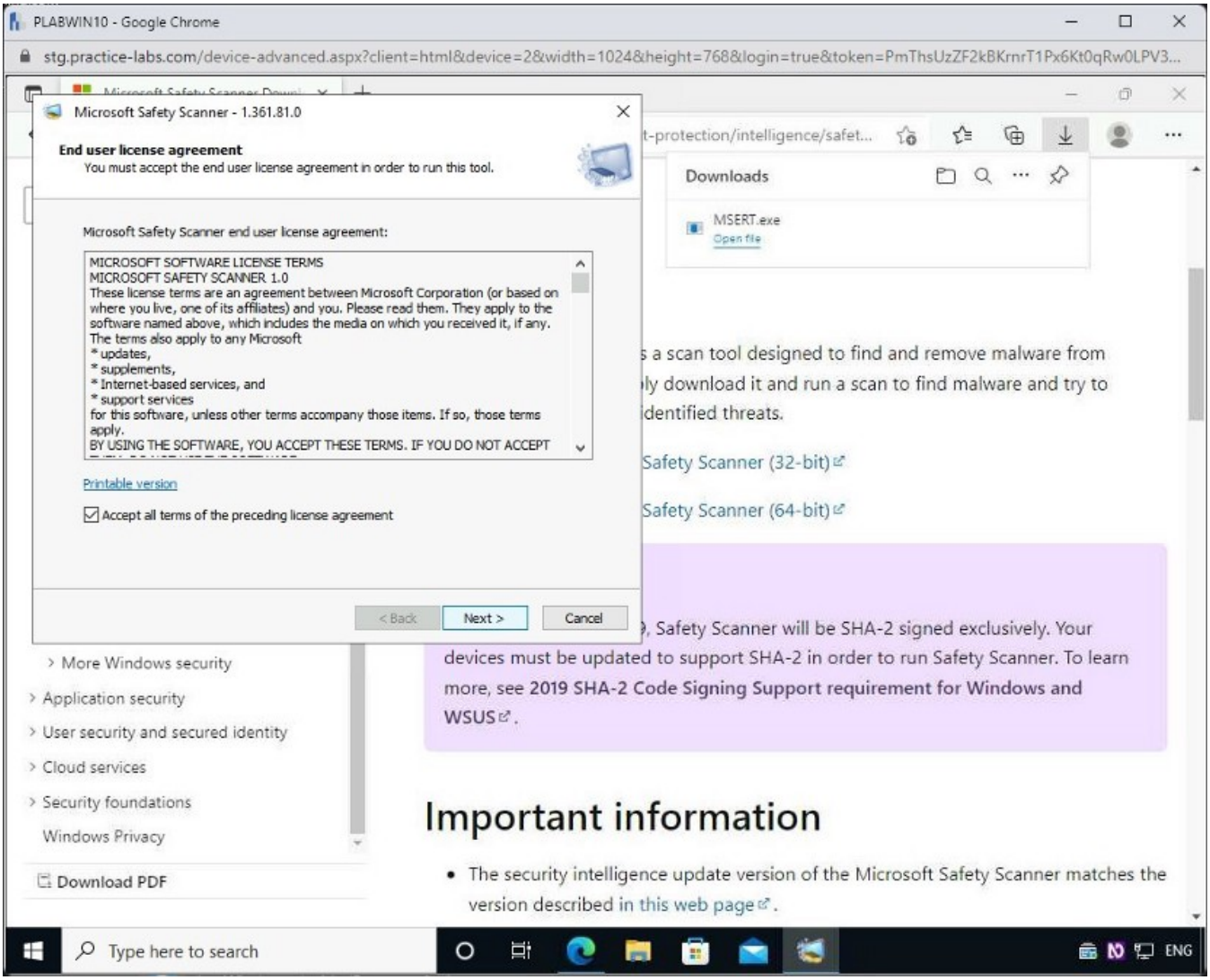
When the notification bar appears on the top of the screen, click **Open file**.



Step 5

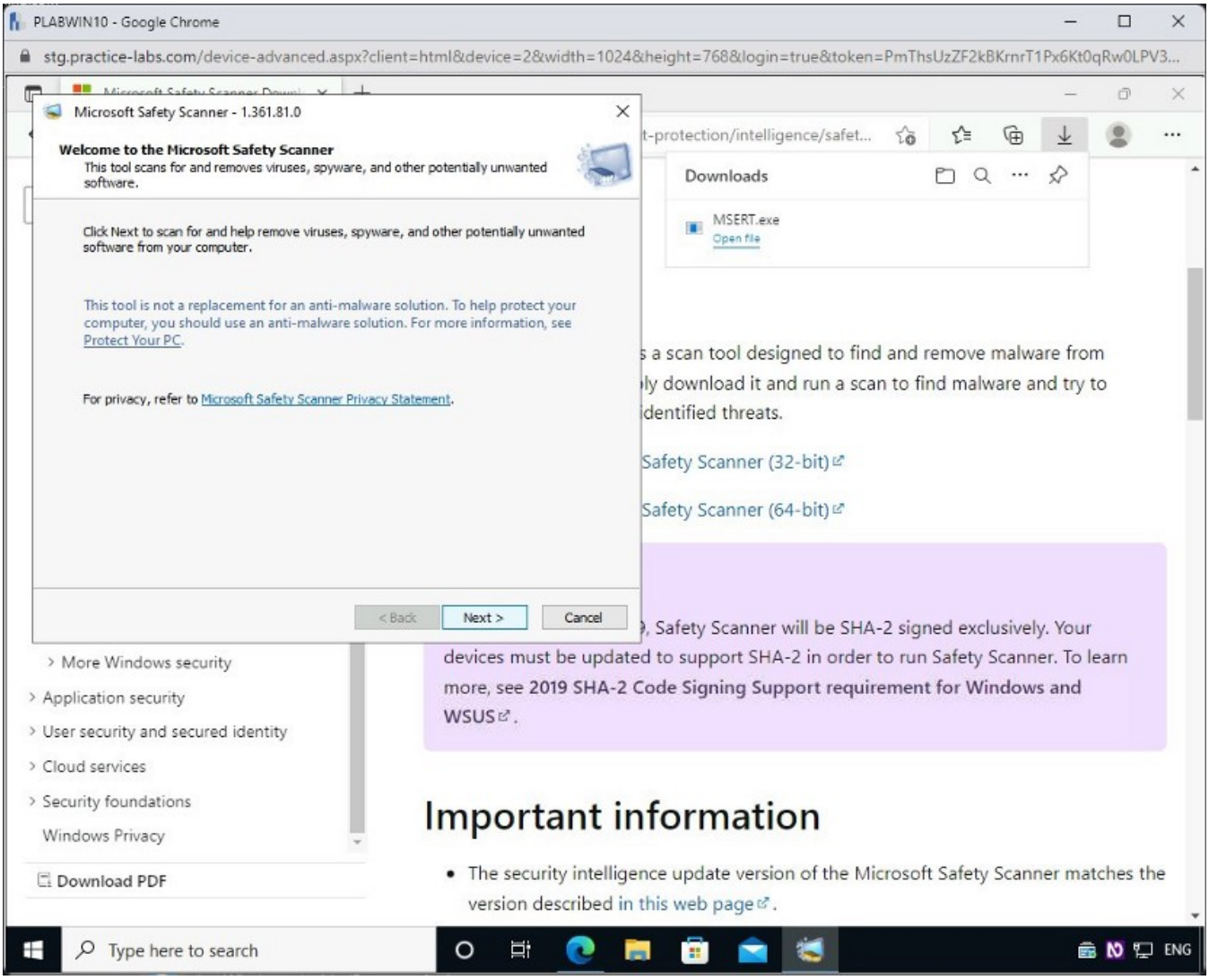
After the download is finished, the **Microsoft Safety Scanner — 1.355.2141.0** dialog box is displayed.

Click the **Accept all terms of the preceding license agreement** checkbox and click **Next**.



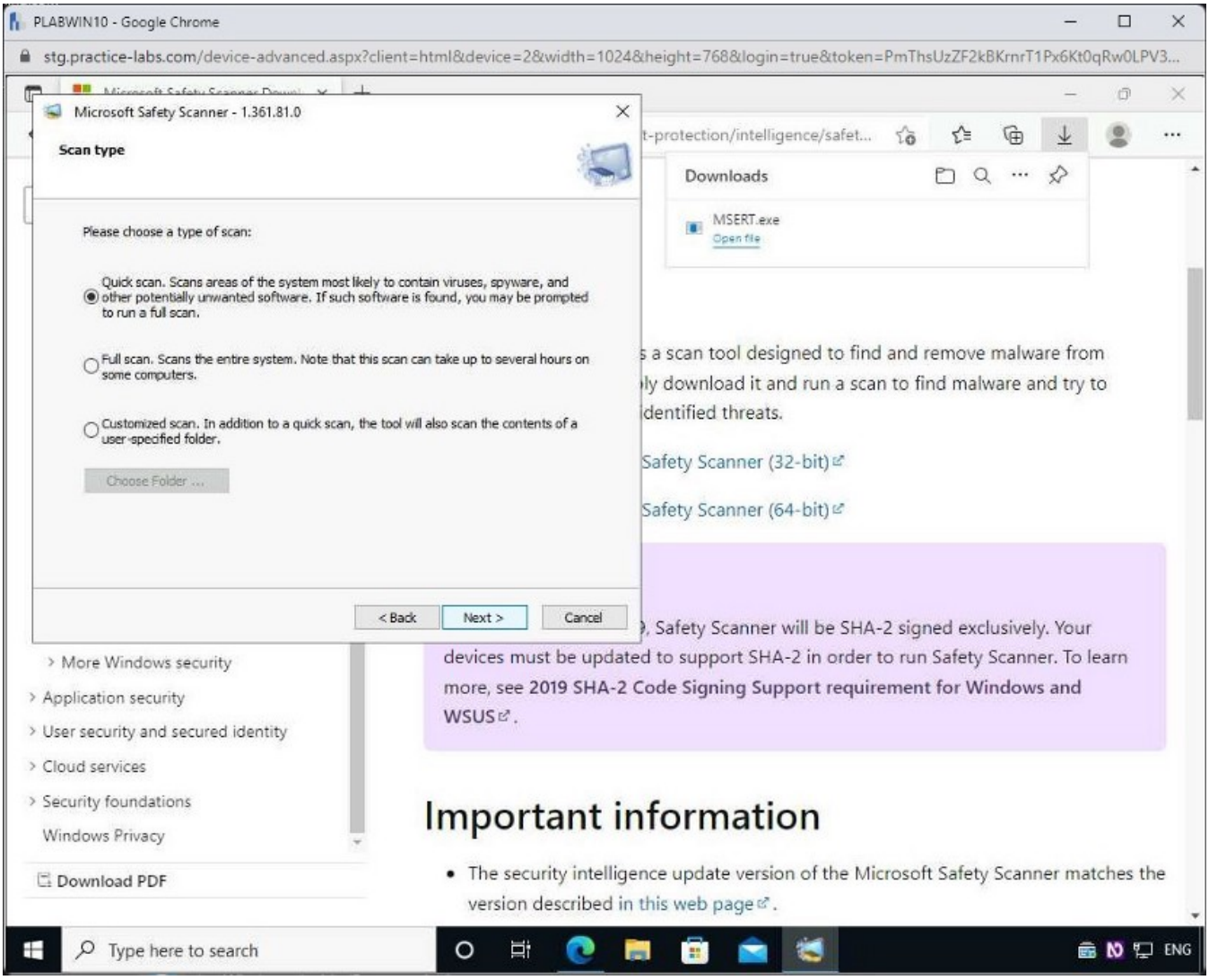
Step 6

In **Welcome to the Microsoft Safety Scanner**, click **Next.**



Step 7

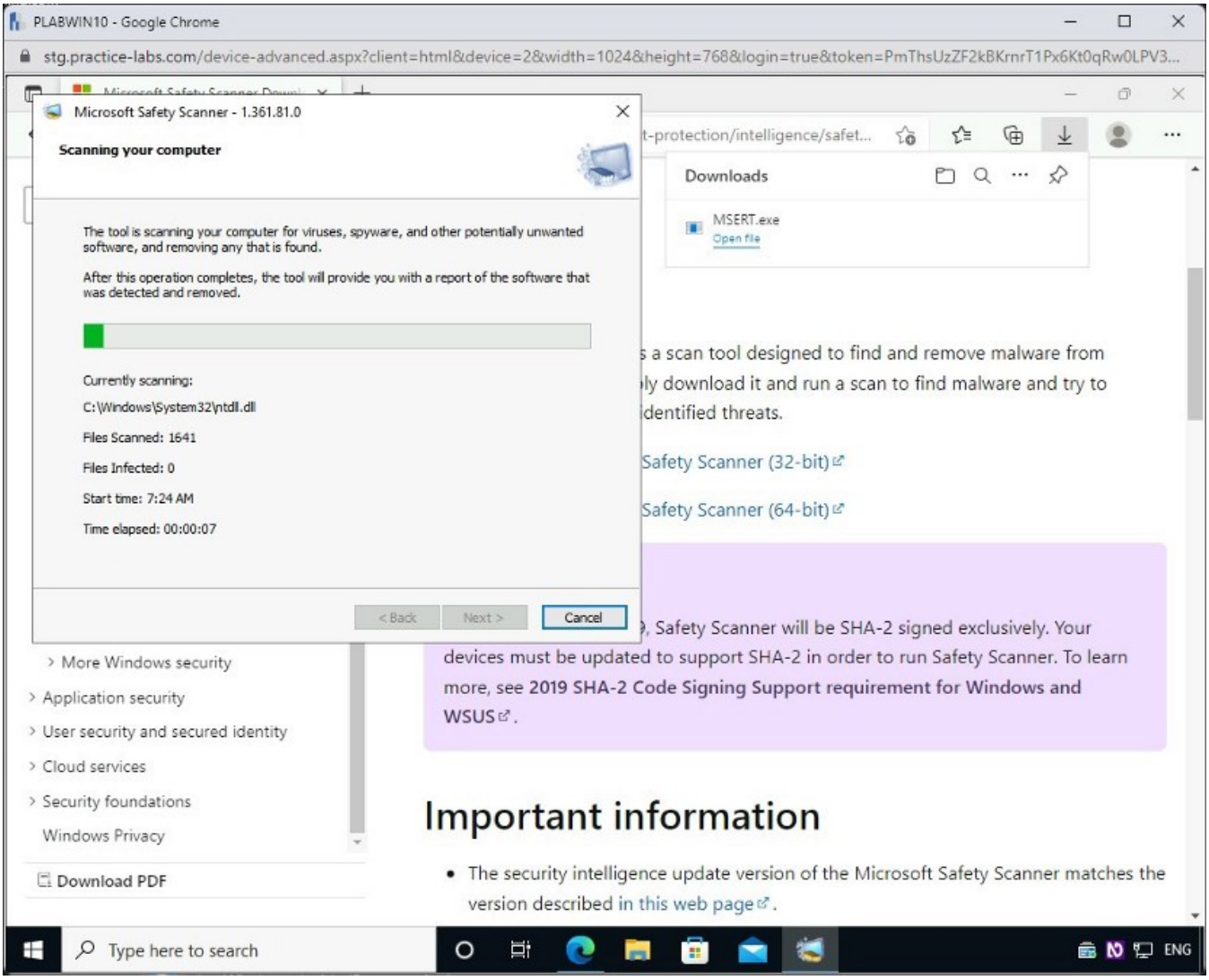
In the **Scan Type** page, keep the default selection of **Quick scan** and then click **Next**.



Step 8

On the **Scanning your computer** page, the scanning progress is displayed.

This will take about 3–4 minutes to complete.

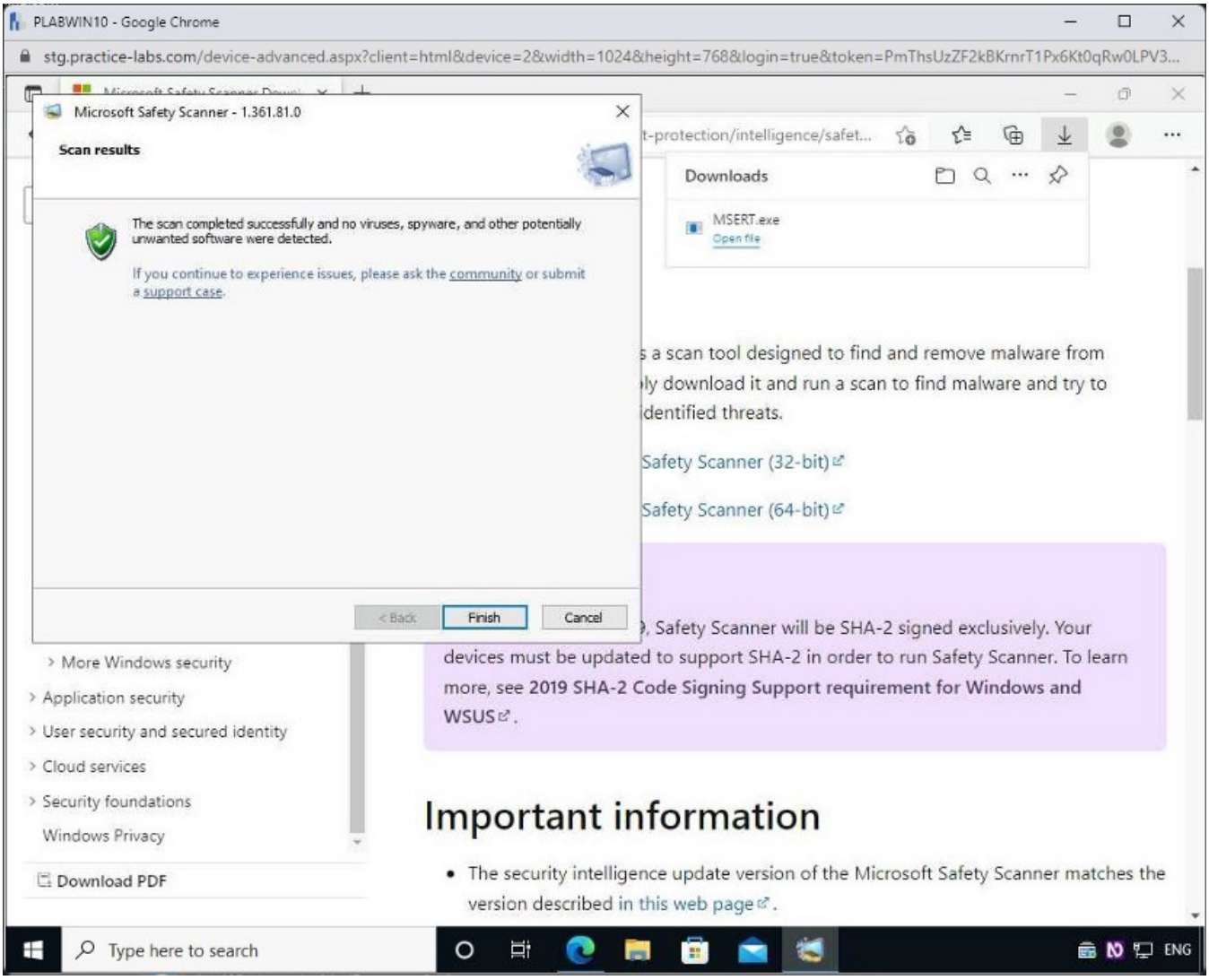


Step 9

After the scan is completed on the **Scan results** page, click **View detailed results of the scan**.

Notice that the scan has located no infected files.

Click **Finish**.



Close the **Microsoft Edge** window.

Task 3 — Use SUPERAntiSpyware

Spyware is a type of malware that silently sits inside your system and collects information. An anti-spyware program is designed to catch spyware in your system, which cannot be caught with a usual anti-virus program. There are different activities that spyware can perform, some of which are installing additional software without your knowledge or redirecting your Web browser to an unwanted website.

An anti-spyware can detect the spyware using the rule-based methods or the latest definition files. An anti-spyware can be an independent program or part of a security suite.

In this task, you will use SuperAntiSpyware. To do this, perform the following steps:

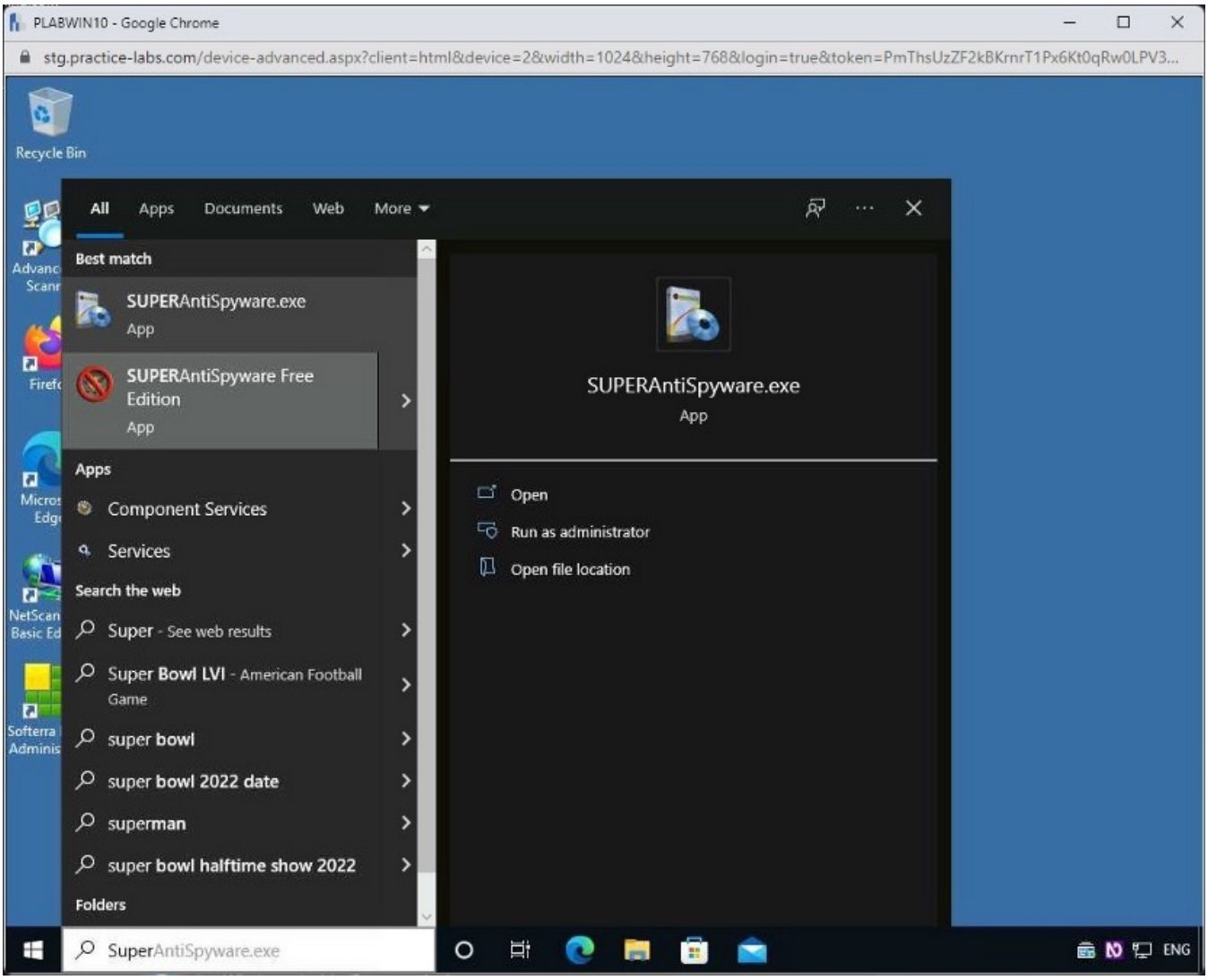
Step 1

Ensure that you are connected to **PLABWIN10**.

In the **Type here to search** textbox, type the following:

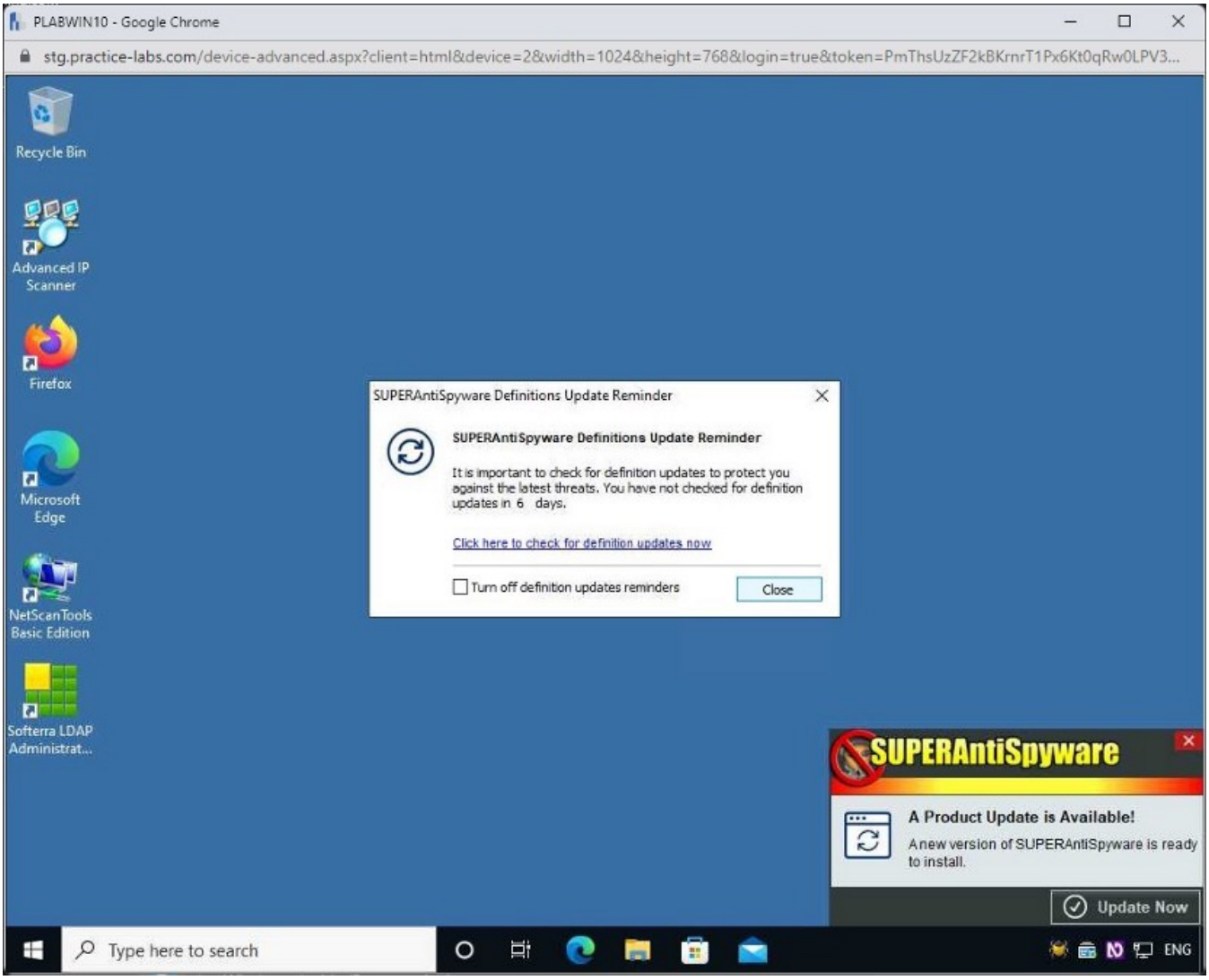
```
SUPERAntiSpyware
```

Click **SUPERAntiSpyware Free Edition** from the search results.



Step 2

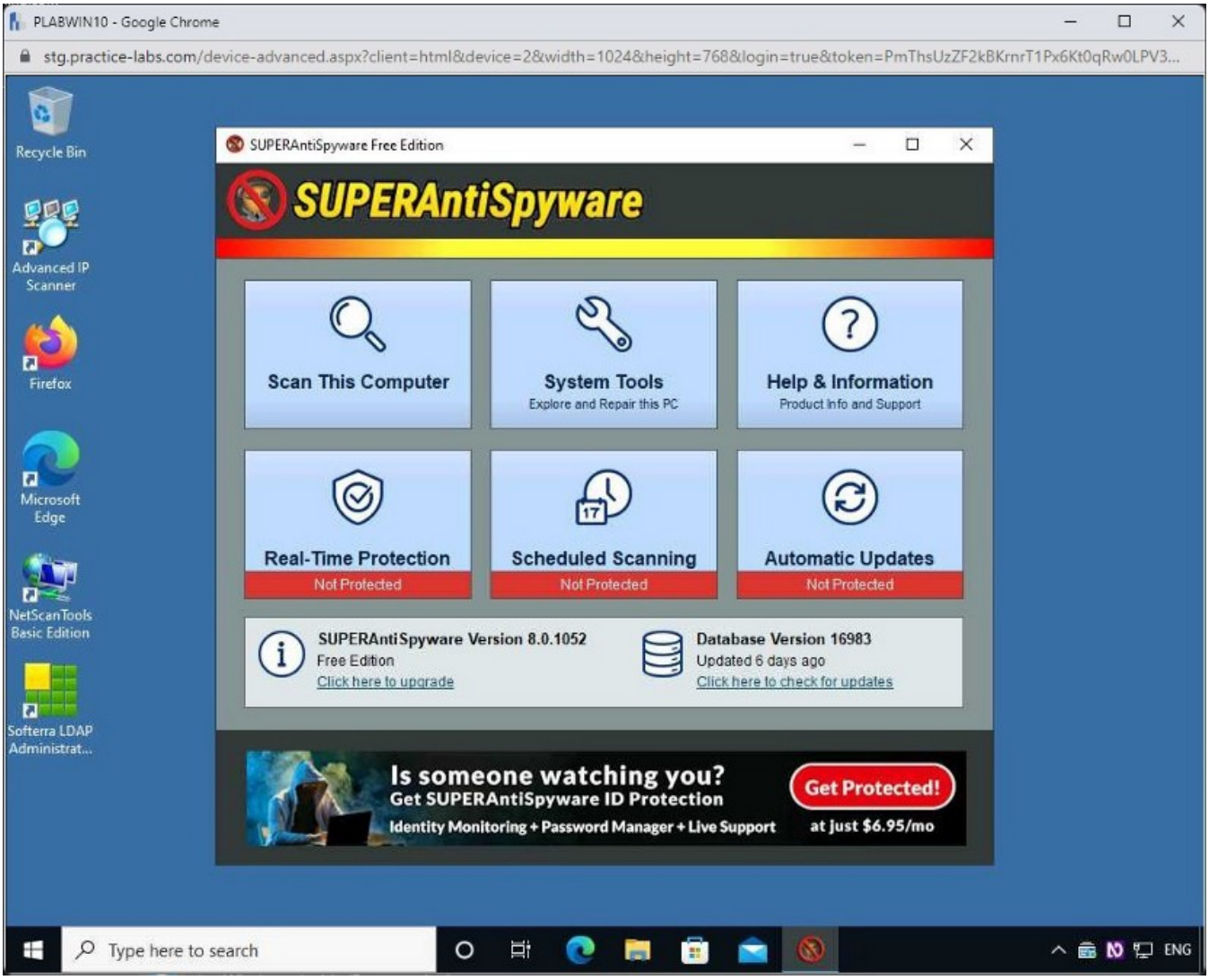
In the SUPERAntiSpyware Defenitions Update Reminder window, click Close.



Step 3

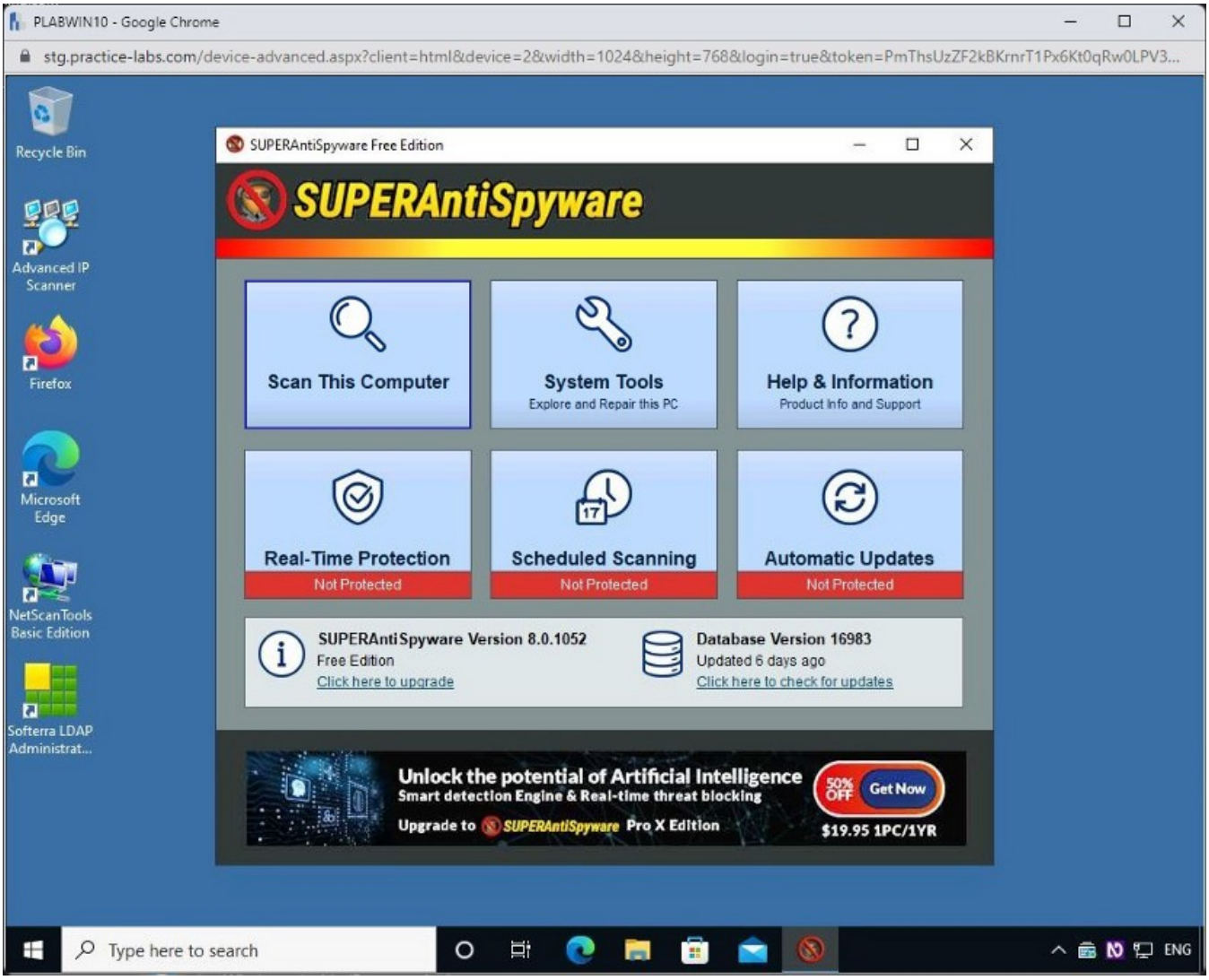
Once again open **SUPERAntiSpyware** via the **Type here to search** textbox.

The program will now open.



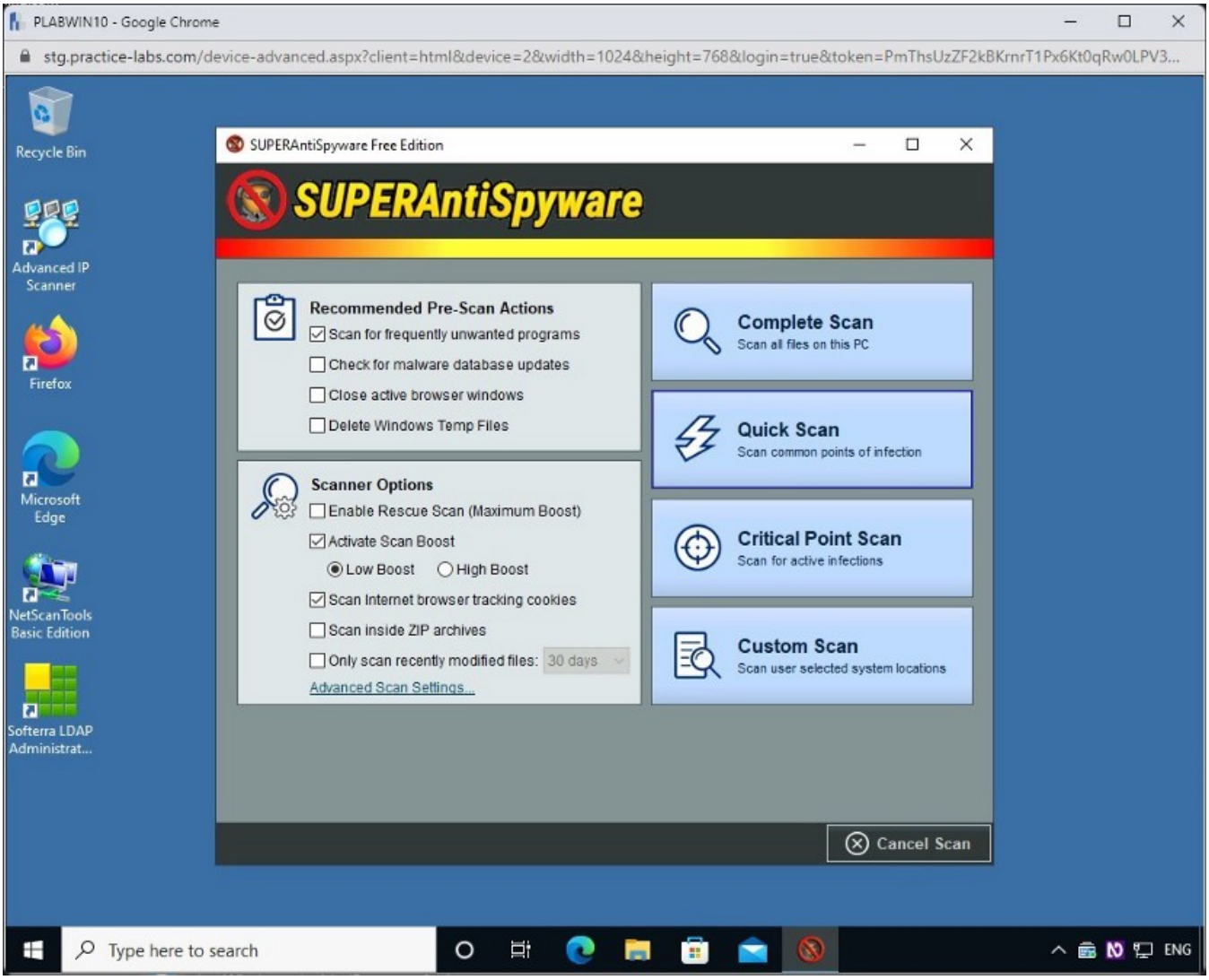
Step 4

Click **Scan This Computer**.



Step 5

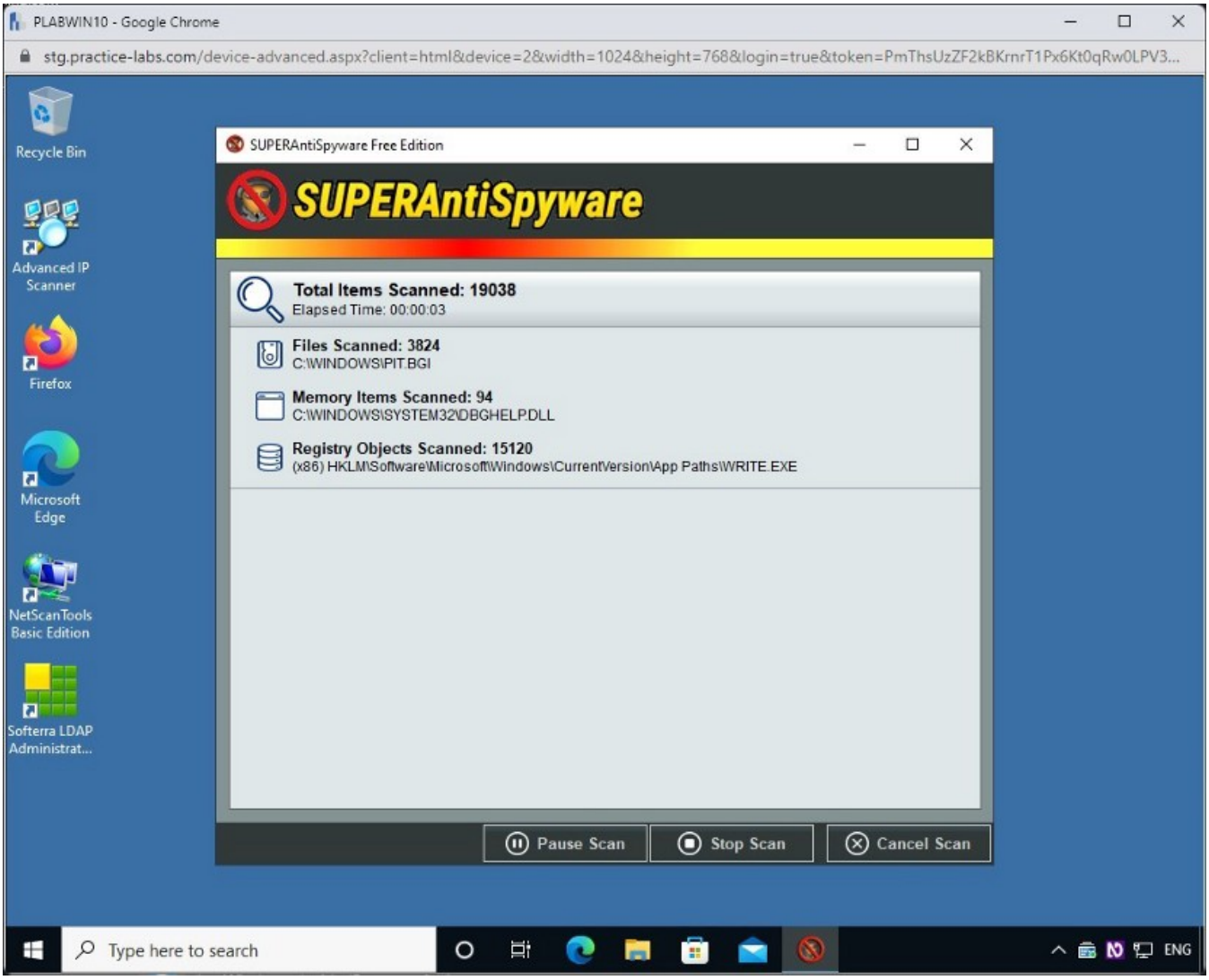
On the scan screen, keep the default options and click **Quick Scan**.



Step 6

The **Quick Scan** will begin.

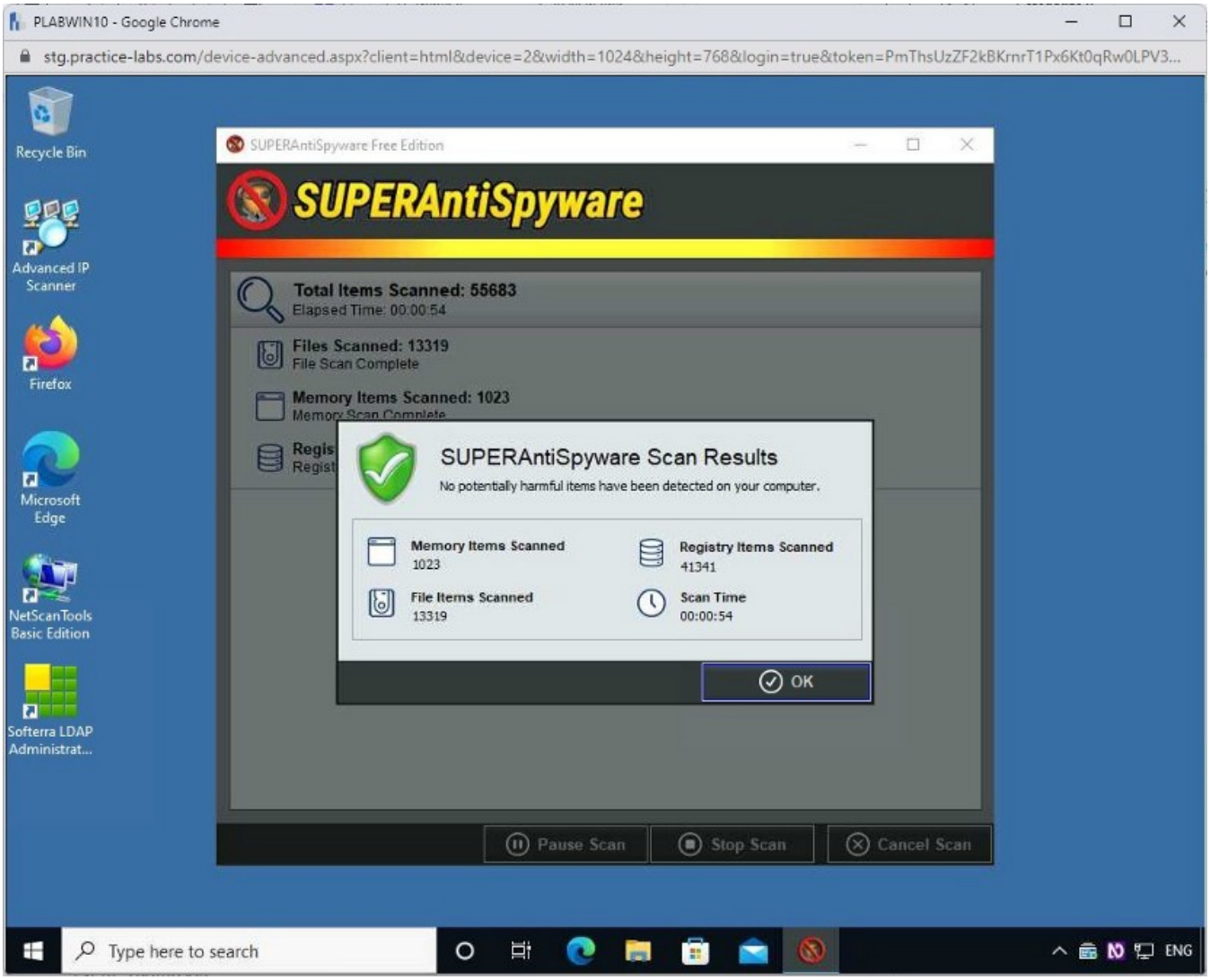
Note: *It may take a while for the scan to finish.*



Step 7

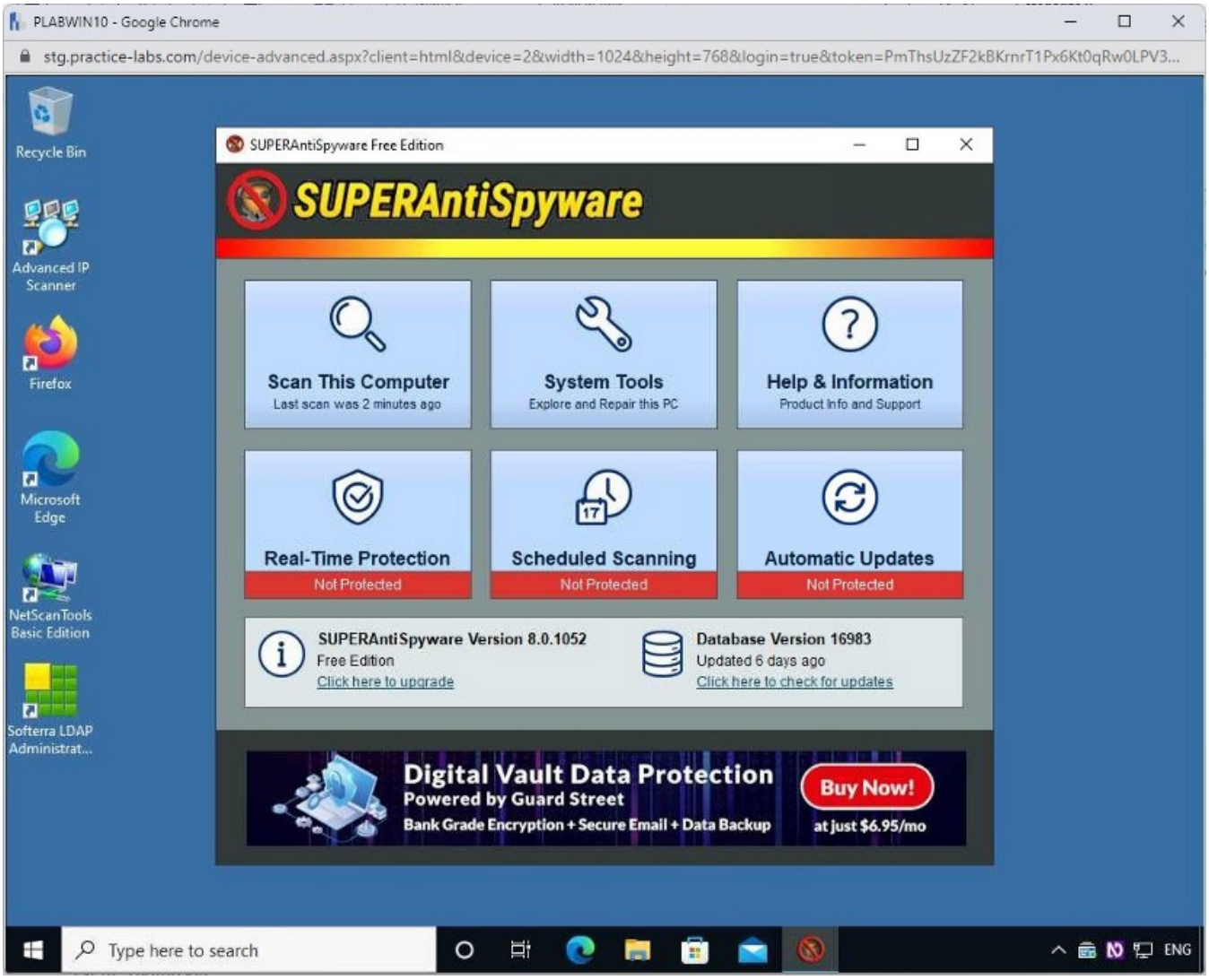
After the scan is finished, the scan results are displayed.

Click **OK**.



Step 8

You are back on the main screen of **SUPERAntiSpyware**.



Exercise 4 — Malware Analysis & Countermeasures

In many cases, malware is the source of an incident. The nature of the incident depends on the malware's capabilities, which cannot be determined unless you reverse engineer it. When you reverse engineer malware, you will see the code's functionality.

With the help of the malware analysis method, you can understand the core functionalities and capabilities of the malware. There should be capabilities with reverse engineering methods and processes to handle malware in an incident.

In this exercise, you will learn about some of the key malware analysis techniques and countermeasures.

Learning Outcomes

After completing this exercise, you will have further knowledge of:

- The objective of Malware Analysis

- Types of Malware Analysis
- Malware Detection Methods
- Malware Countermeasures

Objective of Malware Analysis

Malware (malicious software) is usually complicated in nature. To stop malware and prepare its antidote, you need to know what is hidden inside it, which can be done through malware analysis.

Malware analysis is the process of reverse-engineering malware so you can determine the origin, functionality, and impact the malicious software will have. This is an important part of penetration testing.

Some of the key reasons to perform this analysis include:

- Know the intention of malware
- Understand the vulnerability it targets
- Know its evasion method
- Understand its triggering method
- Know the damage it can cause
- Understand its indicators of compromise

Types of Malware Analysis

Malware analysis is of two types — static and dynamic.

- **Static**: The malware code is analyzed without executing it. Static analysis intends to understand the malware functionality and key attributes like hashes, file size, and file type. Static analysis is also known as code analysis. You can use one of the following static methods to conduct malware analysis:
- Performing file fingerprinting
- Using file obfuscation methods
- Locating file dependencies

- Performing malware de-assembly
- Conducting string search
- Locating the portable executables (PE) methods
- Conducting online and offline malware scanning
- **Dynamic**: In dynamic analysis, the malware is executed to understand its behavior and interaction methods with the system or application. The dynamic analysis, also known as behavioral analysis, is used to understand the key malware attributes, such as installation location, generated registry keys, and dependent files like DLL files. When conducting dynamic analysis, you need to ensure the following:
- Create a system baseline: this will help you understand the system state before and after malware execution
- Host integrity monitoring will help you monitor the current system behavior and the changes after malware execution. For example, the malware opens several ports to send out data.

In both types, care must be taken to ensure that the malware does not propagate beyond the testing computer or network. This is often done by air-gapping a single testing computer, or as is often the case in dynamic analysis, the testing network. Air-gapping of this type for this specific purpose can be referred to as sheep dipping.

Malware Detection Methods

Malware is only effective if it is triggered. Various methods can be used to detect malware before its execution. Some of the key methods are:

- **Scanning**: to protect systems from malware, users should always have an updated version of an anti-malware application. It is necessary to have anti-malware on their system and keep it updated all the time. When new malware is released, the anti-malware application vendors keep releasing the signatures, which should be updated on the systems.
- **Integrity Checking**: integrity checker applications keep track of the files that exist on the system. Each file is marked with a specific signature. If the file is altered, the signature changes, which generates an alert for the user.

- **Interception**: interception applications are interceptors that monitor the requests to the operating systems and network. If there is a suspicious request, it generates an alert for the user.
- **Code Emulation**: code emulation method executes the virus in a virtual machine rather than the real system. In this method, the malware continues to execute until it is detected. The host machine is not impacted. The code emulation method is typically used with malware that uses encryption.
- **Heuristic Analysis**: this method is used for new malware that is released. Both static and dynamic analysis can be used. The code emulation method is used for code execution in the dynamic analysis.

Malware Countermeasures

Eradicating malware from a system can be a challenging task. Therefore, the better solution is to protect the system beforehand to avoid infections.

Performing certain countermeasures can protect systems from malware.

Viruses and worms can be dangerous for systems and networks. Therefore, it is better to apply countermeasures than to go in the recovery mode, which can be time-consuming. Some of the key countermeasures are:

- Avoid opening attachments received from unknown email senders
- Keep your system and applications updated
- Use SPAM applications and filters
- Scan all incoming emails
- Use legitimate programs from known vendors
- Avoid installing pirated applications
- Use a trustworthy anti-malware application from a trusted vendor
- Avoid installing applications from the Internet
- Enable firewall on all systems within a network
- Monitor the incoming and outgoing traffic

- Use registry monitoring tools
- Delete registry entries, files, and executables of the backdoor program
- Disable Windows PowerShell
- Disable Windows Management Instrumentation (WMI)
- Allow only digitally signed macros
- Disable Adobe Flash in Web browsers
- Disable PDF readers to execute JavaScript automatically
- Configure multi-factor authentication (MFA) and User Behavior Analytics (UBA) tools on systems
- Use layered defense — IDS/IPS, firewall, and anti-malware
- Use only limited Web browser add-ons from trusted providers
- Create a system baseline and monitor any changes to the baseline
- Use Microsoft Enhanced Mitigation Toolkit
- Remove unnecessary applications
- Filter and monitor the incoming and outgoing traffic
- Keep the anti-malware updated — all the time
- Regularly scan your system with the updated anti-malware application
- Execute only whitelisted applications
- Apply operating system and application updates regularly
- Work with the principle of least privileges
- Review logs from time to time
- Backup your data regularly
- Disable unnecessary services and open ports