INTRUVENT

COMPREHENSIVE THREAT HUNTING GUIDE

APT34 / OilRig: DNS Tunneling Detection and Sleeper Access Discovery

Detecting DNS-based C2 channels, hunting OilRig custom tooling, identifying dormant implants, and detecting CVE-2024-30088 exploitation

Version 1.0
Classification TLP:CLEAR
Published March 18, 2026
Total Hunt Time 5 Hours
Threat Actor APT34 / OilRig / Hazel Sandstorm (G0049)
Target Sectors Government, Financial Services, Energy, Telecommunications

Intruvent Technologies
INT-THG-2026-APT34-DNS-v1.0

Table of Contents

Overview

Executive Summary 3
MITRE ATT&CK Mapping 4

Hunt Modules

Module 1: DNS Tunneling Detection 5
Module 2: OilRig Custom Tool Hash and Behavioral Detection 6
Module 3: CVE-2024-30088 Windows Kernel Exploitation Detection 7
Module 4: Credential Harvesting via Mimikatz/Custom Dumpers 8
Module 5: Scheduled Task Persistence Patterns 9
Module 6: Web Shell Detection (Exchange/IIS) 10
Module 7: Dormant Implant Discovery 11
Module 8: Lateral Movement via Pass-the-Hash/WMI 12
Module 9: Data Staging and Exfiltration via DNS 13
Module 10: Email Account Compromise and Mailbox Export Detection 14

Appendices

Appendix A: Indicators of Compromise 15
Appendix B: Detection Rules 17
Appendix C: References 19

1 Executive Summary

AT A GLANCE

THREAT ACTOR
APT34 / OilRig / Hazel Sandstorm (G0049, Iran)

PRIMARY OBJECTIVE
Long-duration espionage via DNS tunneling C2 and email-based exfiltration

ACTIVE SINCE
2014 (DNS tunneling C2 since at least May 2016)

TARGET SECTORS
Government, Financial Services, Energy, Telecommunications, Defense

KEY DIFFERENTIATOR
DNS tunneling as signature C2 channel across 6+ backdoor families

RECENT ACTIVITY
CVE-2024-30088 exploitation in UAE (2024); dormant infrastructure (2025)

APT34 (also tracked as OilRig by Palo Alto Unit 42, Hazel Sandstorm by Microsoft, Helix Kitten by CrowdStrike, Earth Simnavaz by Trend Micro, Crambus by Symantec, COBALT GYPSY by Secureworks, MITRE G0049) is an Iranian state-sponsored cyber espionage group active since at least 2014. The group targets Middle Eastern governments, financial institutions, energy companies, and telecommunications providers for intelligence collection on behalf of the Iranian state.

APT34's most distinctive operational characteristic is its sustained reliance on DNS tunneling for command-and-control communication. Since at least May 2016, the group has deployed six or more backdoor families (Helminth, ISMAgent, ALMA Communicator, BONDUPDATER, QUADAGENT, Saitama) that encode commands and data within DNS queries and responses. DNS traffic is almost universally permitted through firewalls, making this channel difficult to block without disrupting legitimate services. The tradeoff is that DNS tunneling generates high query volumes, creating a detectable signal for defenders with DNS analytics capabilities.

This guide provides **10 hunt modules** designed to help defenders: (1) detect DNS tunneling C2 channels used by APT34, (2) identify known OilRig custom tools on endpoints, (3) discover CVE-2024-30088 exploitation artifacts, (4) find dormant implants and sleeper access mechanisms, and (5) detect credential harvesting and email-based exfiltration patterns. Total estimated hunt time is **5 hours**.

Threat Overview

APT34 represents one of Iran's most persistent and technically capable cyber espionage operations. The group has demonstrated sustained operational capability across more than a decade, surviving a major tool leak in April 2019 by rapidly developing an entirely new malware arsenal. This resilience signals strong state backing and dedicated development resources.

The group's operational tempo accelerated in 2024 with the Earth Simnavaz campaign targeting UAE government entities, which introduced CVE-2024-30088 exploitation (a Windows kernel TOCTOU race condition) and the STEALHOOK exfiltration tool. By late 2024 and into 2025, APT34 was observed pre-positioning infrastructure impersonating Iraqi academic institutions and fictitious UK-based technology firms, suggesting preparation for future campaigns.

APT34 has evolved its C2 strategy in recent years, supplementing DNS tunneling with email-based C2 (PowerExchange, MrPerfectionManager, STEALHOOK) and cloud service-based downloaders (SampleCheck5000, OilCheck, OilBooster, ODAgent) that use Microsoft 365 and OneDrive APIs. This multi-channel approach provides redundancy and makes detection more complex.

KEY CHARACTERISTICS

- Iranian state-sponsored (assessed IRGC-linked)
- DNS tunneling C2 (signature technique)
- 18+ custom backdoor families
- Email-based C2 via Exchange
- Cloud service C2 (M365, OneDrive)
- CVE-2024-30088 exploitation
- Password filter DLL persistence
- Web shell deployment (IIS/Exchange)
- ngrok for reverse tunneling
- Supply chain compromise capability

2 MITRE ATT&CK Mapping

The following techniques are mapped from MITRE ATT&CK G0049, Intruvent Codex (75 techniques), and vendor reporting from Trend Micro, ESET, Check Point, Unit 42, and Symantec:

T1071.004 SIGNATURE
Application Layer Protocol: DNS
Primary C2 channel across Helminth, ISMAgent, BONDUPDATER, QUADAGENT, Saitama. High-entropy subdomains encode commands/data.

T1071.001
Application Layer Protocol: Web Protocols
HTTP/HTTPS C2 used by SideTwist, OopsIE, Solar. Fake 404 pages as response masking.

T1048.003
Exfiltration Over Unencrypted/Obfuscated Protocol
Data exfiltrated via DNS queries, email attachments through Exchange, and steganography (RDAT BMP images).

T1068 2024
Exploitation for Privilege Escalation
CVE-2024-30088 Windows kernel TOCTOU race condition exploited in Earth Simnavaz UAE campaign.

T1556.002
Modify Authentication: Password Filter DLL
Password filter DLL registered to intercept credentials. Used to deploy STEALHOOK backdoor.

T1505.003
Web Shell
SEASHARPEE and custom web shells on IIS/Exchange for persistent access and fallback C2.

T1053.005
Scheduled Task
Scheduled tasks for persistence; used across multiple backdoor families for periodic C2 check-in.

T1003.001
OS Credential Dumping: LSASS Memory
Mimikatz and LaZagne used for credential harvesting from LSASS memory.

T1047
Windows Management Instrumentation
WMI used for lateral movement and remote command execution.

T1572
Protocol Tunneling
ngrok reverse proxy tunneling for persistent access through firewalls.

T1566.001
Spearphishing Attachment
Primary initial access vector. Malicious Office documents with macros delivering backdoors.

T1219
Remote Access Software
ngrok used as legitimate remote access tool for tunneling in Earth Simnavaz campaign.

3 Hunt Modules

1 DNS Tunneling Detection

Duration: 40 min Priority: CRITICAL

Objective: Detect DNS tunneling C2 channels characteristic of APT34 backdoors by identifying high-entropy subdomain queries, excessive TXT record lookups, and anomalous DNS query volumes from individual endpoints.

Background

APT34 has used DNS tunneling for C2 communication since at least May 2016 across Helminth, ISMAgent, ALMA Communicator, BONDUPDATER, QUADAGENT, and Saitama backdoors. The tunneling protocol encodes data in subdomain labels of DNS queries. Each query includes a randomly generated component to avoid cached responses. The C2 server responds with hardcoded IP addresses that serve as control signals (start/stop data transfer, acknowledge receipt). An initial handshake obtains a unique system identifier for session tracking. (Source: Palo Alto Unit 42, "DNS Tunneling in the Wild: Overview of OilRig's DNS Tunneling")

Hunt Procedure: High-Entropy Subdomain Detection (KQL)

DETECTION ONLY

KQL (MICROSOFT SENTINEL)

```
// Detect DNS queries with abnormally long subdomain labels
// APT34 DNS tunneling encodes data in subdomains, producing labels >50 chars
// Source: Unit 42 OilRig DNS tunneling analysis
DnsEvents
| where Name has "."
| extend SubdomainLabel = tostring(split(Name, ".")[0])
| extend LabelLength = strlen(SubdomainLabel)
| where LabelLength > 50
| summarize
    QueryCount = count(),
    DistinctSubdomains = dcount(Name),
    SampleQueries = make_set(Name, 5)
    by ClientIP, tostring(split(Name, ".")[-2]),
    bin(TimeGenerated, 1h)
| where QueryCount > 100 and DistinctSubdomains > 50
| sort by QueryCount desc
```

Hunt Procedure: Excessive TXT Record Queries (KQL)

DETECTION ONLY

KQL (MICROSOFT SENTINEL)

```
// OilRig backdoors use TXT, A, and AAAA records for data transfer
// TXT records carry larger payloads per query
DnsEvents
| where QueryType in ("TXT", "16")
| summarize
    TxtQueryCount = count(),
    UniqueDomains = dcount(Name)
    by ClientIP, bin(TimeGenerated, 1h)
| where TxtQueryCount > 200
| sort by TxtQueryCount desc
```

Hunt Procedure: DNS Query Volume Anomaly (SPL)

DETECTION ONLY

SPL (SPLUNK)

```
`dns`
| eval subdomain_len=len(mvindex(split(query,"."),0))
| where subdomain_len > 50
| eval query_domain=mvindex(split(query,"."), -2) . "." . mvindex(split(query,"."), -1)
| stats sum(subdomain_len) as total_bytes, count as query_count, dc(query) as unique_queries,
    values(query) as sample_queries by src_ip, query_domain
| where query_count > 100 AND unique_queries > 50
| sort -query_count
```

- ☐ Baseline normal DNS query volumes per endpoint before running these queries
- ☐ Whitelist known-legitimate high-volume DNS domains (CDNs, AV updates, cloud services)
- ☐ Focus on queries to uncommon or recently registered domains
- ☐ Check if flagged domains resolve to hardcoded IP addresses (APT34 control signal pattern)
- ☐ Correlate DNS anomalies with process creation events on the source endpoint

False Positive Guidance
CDN providers, anti-malware update services, DKIM verification, and cloud applications generate high-volume DNS queries with long labels. Microsoft 365, Google Workspace, and Akamai are common false positive sources. Build a whitelist of known-good high-volume DNS domains before deploying these queries in production.

Escalation Criteria
Any endpoint generating >500 DNS queries per hour to a single domain with subdomain labels exceeding 50 characters, where the domain is not in your whitelist, warrants immediate investigation. Isolate the endpoint and collect a full DNS query log for the prior 30 days.

2 OilRig Custom Tool Hash and Behavioral Detection

Duration: 30 min Priority: CRITICAL

Objective: Detect known APT34 custom backdoors (SideTwist, Saitama, RDAT, Helminth, Solar, Mango, PowerExchange, STEALHOOK) on endpoints using verified file hashes and behavioral indicators.

Background

APT34 maintains an extensive custom malware arsenal spanning 18+ backdoor families. After the April 2019 tool leak, the group rapidly developed replacement tools (SideTwist, Saitama, MrPerfectionManager, Solar/Mango family). Hash-based detection provides the highest-confidence identification. Behavioral detection catches variants that have been recompiled with new hashes.

Hunt Procedure: Known OilRig File Hashes (KQL)

BLOCKING SAFE

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect known APT34 malware hashes on endpoints
// Sources: NSFOCUS (2023), Check Point (2021), Joe Sandbox (2024)
let OilRigHashes = dynamic([
    "47d3e6c389cfdbc9cf7eb61f3051c9f4e50e30cf2d97499144e023ae87d68d5a",  // SideTwist backdoor
    "13c27e5049a7fc5a36416f2c1ae49c12438d45ce50a82a96d3f792bfdacf3dcd",  // SideTwist delivery doc
    "704360d765f6f1ef735594c7ff5fb6c47467dad8abc3133f8e935a6c0c804c8a"   // APT34 malware sample (2024)
]);
DeviceFileEvents
| where SHA256 in (OilRigHashes)
| project Timestamp, DeviceName, FileName, FolderPath, SHA256,
    ActionType, InitiatingProcessFileName
```

Hunt Procedure: Saitama PDB Path Detection (KQL)

BLOCKING SAFE

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Search for the Saitama backdoor PDB path in process metadata
// Source: Malwarebytes/Fortinet (2022)
DeviceProcessEvents
| where ProcessCommandLine has_any (
    "Saitama.Agent",
    "E:\\Saitama\\")
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, InitiatingProcessFileName
```

Hunt Procedure: OilRig Behavioral Indicators (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect behavioral patterns common across OilRig backdoors
// Focus on .NET processes making DNS queries and spawning cmd.exe
DeviceProcessEvents
| where InitiatingProcessFileName in~ ("w3wp.exe", "powershell.exe")
| where FileName in~ ("cmd.exe", "nslookup.exe")
| where ProcessCommandLine has_any (
    "nslookup", "txt", "set type=")
| project Timestamp, DeviceName, InitiatingProcessFileName,
    FileName, ProcessCommandLine, AccountName
| sort by Timestamp desc
```

- ☐ Run hash lookups against all endpoint telemetry for the past 90 days minimum
- ☐ Obtain additional hashes from vendor reports listed in Appendix A
- ☐ Any hash match warrants immediate endpoint isolation
- ☐ Check VirusTotal for hash matches to confirm detection rates

Critical: Any Hash Match Is High Confidence
All hashes in this module are sourced from published vendor research. A match on any hash warrants immediate endpoint isolation and forensic investigation. These are confirmed APT34 tools with no legitimate use case.

3 CVE-2024-30088 Windows Kernel Exploitation Detection

Duration: 25 min Priority: CRITICAL

Objective: Detect exploitation of CVE-2024-30088 (Windows Kernel TOCTOU race condition) as used by APT34/Earth Simnavaz for privilege escalation, and identify downstream artifacts including password filter DLL registration and STEALHOOK deployment.

Background

In the 2024 Earth Simnavaz campaign against UAE government entities, APT34 exploited CVE-2024-30088 (patched by Microsoft in June 2024) to escalate privileges. The exploit binary was loaded into memory using the open-source RunPE-In-Memory tool. After achieving SYSTEM privileges, the group registered a password filter DLL that intercepted domain credentials and deployed the STEALHOOK exfiltration tool to send stolen data through Exchange servers. (Source: Trend Micro, October 2024)

Hunt Procedure: Password Filter DLL Registration (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect new password filter DLL registrations
// APT34 registers malicious DLLs to intercept domain credentials
// Source: Trend Micro Earth Simnavaz report (Oct 2024)
DeviceRegistryEvents
| where RegistryKey has @"SYSTEM\CurrentControlSet\Control\Lsa"
| where RegistryValueName == "Notification Packages"
| where ActionType == "RegistryValueSet"
| project Timestamp, DeviceName, RegistryKey,
    RegistryValueName, RegistryValueData,
    InitiatingProcessFileName, InitiatingProcessCommandLine
| sort by Timestamp desc
```

Hunt Procedure: RunPE-In-Memory Indicators (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect process hollowing patterns consistent with RunPE-In-Memory
// APT34 loads exploit binaries into memory using this technique
DeviceProcessEvents
| where ProcessCommandLine has_any (
    "RunPE", "InMemory", "Hollow")
    or (FileName in~ ("svchost.exe", "explorer.exe")
        and InitiatingProcessFileName in~ ("powershell.exe", "cmd.exe", "wscript.exe"))
| where InitiatingProcessCommandLine has_any (
    "bypass", "hidden", "encoded", "-e ", "-enc")
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, InitiatingProcessFileName,
    InitiatingProcessCommandLine, AccountName
```

Hunt Procedure: Patch Verification (PowerShell)

INVESTIGATION AID

POWERSHELL

```
# Check if CVE-2024-30088 patch (KB5039211 or later) is installed
# June 2024 Patch Tuesday addressed this vulnerability
Get-HotFix | Where-Object {
    $_.HotFixID -in @("KB5039211","KB5039212","KB5039213",
                      "KB5039214","KB5039217")
} | Select-Object HotFixID, InstalledOn, Description
```

- ☐ Verify CVE-2024-30088 patch status across all Windows domain controllers and servers
- ☐ Review password filter DLL registrations for any non-standard entries
- ☐ Check for unsigned DLLs in the LSA notification packages list
- ☐ Investigate any RunPE-In-Memory artifacts or process hollowing events

Escalation Criteria
Any new password filter DLL registration is high priority. Legitimate password filter DLLs are rare outside of password policy enforcement products. Any unsigned DLL or DLL from a non-standard path registered as a notification package warrants immediate forensic investigation of the domain controller.

4 Credential Harvesting via Mimikatz/Custom Dumpers

Duration: 30 min Priority: HIGH

Objective: Detect credential harvesting activity consistent with APT34's documented use of Mimikatz (S0002) and LaZagne (S0349) for LSASS memory dumping, LSA secrets extraction, and browser credential theft.

Background

APT34 consistently uses Mimikatz and LaZagne across campaigns for credential access. The Intruvent Codex confirms associations with techniques T1003.001 (LSASS Memory), T1003.004 (LSA Secrets), T1003.005 (Cached Domain Credentials), T1555.003 (Credentials from Web Browsers), and T1555.004 (Windows Credential Manager). Harvested credentials enable lateral movement and are used to access Exchange servers for email-based exfiltration.

Hunt Procedure: LSASS Access Detection (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect non-system processes accessing LSASS memory
// APT34 uses Mimikatz for credential dumping (MITRE T1003.001)
DeviceProcessEvents
| where ProcessCommandLine has_any ("sekurlsa", "logonpasswords", "lsadump", "kerberos::list", "crypto::certificates")
    or (ProcessCommandLine has "comsvcs.dll" and ProcessCommandLine has "MiniDump")
    or (ProcessCommandLine has "procdump" and ProcessCommandLine has "lsass")
    or FileName in~ ("mimikatz.exe", "procdump.exe", "procdump64.exe")
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, InitiatingProcessFileName,
    AccountName
| sort by Timestamp desc
```

Hunt Procedure: LaZagne Execution Detection (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect LaZagne credential recovery tool
// APT34 uses LaZagne for multi-source credential theft
DeviceProcessEvents
| where FileName has "lazagne"
    or (ProcessCommandLine has "lazagne"
        and ProcessCommandLine has_any ("all", "browsers", "sysadmin", "databases", "mails", "wifi"))
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, AccountName
| sort by Timestamp desc
```

Hunt Procedure: Credential Dump via SPL

DETECTION ONLY

SPL (SPLUNK)

```
index=windows sourcetype="WinEventLog:Security" EventCode=4688
| search (NewProcessName="*mimikatz*" OR NewProcessName="*lazagne*"
    OR CommandLine="*sekurlsa*" OR CommandLine="*logonpasswords*"
    OR CommandLine="*procdump*lsass*")
| table _time, ComputerName, SubjectUserName,
    NewProcessName, CommandLine
```

- ☐ Review all LSASS access events for the past 90 days
- ☐ Whitelist legitimate security products that access LSASS (EDR agents, AV solutions)
- ☐ Check for credential dumping tools in temporary directories or user profiles
- ☐ Correlate credential access with subsequent lateral movement events

False Positive Guidance
Legitimate security products (EDR agents, vulnerability scanners, identity protection tools) may access LSASS memory. Windows Defender Credential Guard, if enabled, reduces the attack surface. Focus on non-standard processes accessing LSASS, particularly from user-writable directories or running under non-SYSTEM accounts.

5 Scheduled Task Persistence Patterns

Duration: 25 min Priority: HIGH

Objective: Detect OilRig-specific scheduled task persistence patterns, including tasks that execute PowerShell, VBScript, or custom backdoors at regular intervals for C2 check-in.

Background

APT34 uses scheduled tasks (T1053.005) across multiple backdoor families for persistence. BONDUPDATER, QUADAGENT, and Helminth create scheduled tasks that execute at regular intervals to check in with C2 servers. Tasks often invoke PowerShell or wscript.exe to run VBS scripts that initiate DNS tunneling sessions. The group has been observed naming tasks to blend with legitimate Windows update or maintenance tasks.

Hunt Procedure: Suspicious Scheduled Task Creation (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect scheduled tasks creating persistence for scripts/backdoors
// APT34 uses schtasks for periodic C2 check-in
DeviceProcessEvents
| where FileName =~ "schtasks.exe"
| where ProcessCommandLine has "/create"
| where ProcessCommandLine has_any (
    "powershell", "wscript", "cscript", "cmd.exe",
    "mshta", "certutil", "bitsadmin", ".vbs", ".ps1")
| project Timestamp, DeviceName, ProcessCommandLine,
    AccountName, InitiatingProcessFileName
| sort by Timestamp desc
```

Hunt Procedure: Scheduled Task Audit via SPL

DETECTION ONLY

SPL (SPLUNK)

```
index=windows sourcetype="WinEventLog:Security" EventCode=4698
| spath input=TaskContent
| search (Actions.Exec.Command="*powershell*"
    OR Actions.Exec.Command="*wscript*"
    OR Actions.Exec.Command="*cmd*")
| table _time, ComputerName, SubjectUserName,
    TaskName, Actions.Exec.Command, Actions.Exec.Arguments
```

- ☐ Enumerate all scheduled tasks on domain controllers and Exchange servers
- ☐ Look for tasks running PowerShell or wscript at intervals of 1-30 minutes
- ☐ Check task action paths for scripts in unusual directories (temp, user profiles, ProgramData)
- ☐ Verify task creation timestamps against known administrative change windows

False Positive Guidance
Legitimate IT management tools (SCCM, Intune, patching solutions) create scheduled tasks that invoke PowerShell. Focus on tasks created outside normal change windows, tasks running under non-standard service accounts, and tasks pointing to user-writable directories.

6 Web Shell Detection (Exchange/IIS)

Duration: 30 min Priority: CRITICAL

Objective: Detect SEASHARPEE and other web shells deployed by APT34 on Exchange and IIS servers, which serve as persistent access and fallback C2 channels.

Background

APT34 deploys custom web shells (SEASHARPEE, S0185) and the RGDoor IIS backdoor (S0258) on compromised web servers. In the Earth Simnavaz campaign, the group deployed web shells as the initial post-exploitation persistence mechanism, followed by ngrok tunneling for lateral movement. Web shells provide reliable fallback access when primary C2 channels are disrupted.

Hunt Procedure: New ASPX/ASHX Files on Web Servers (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect new web shell files created on Exchange/IIS servers
// APT34 deploys SEASHARPEE and custom ASPX shells
DeviceFileEvents
| where FileName endswith ".aspx" or FileName endswith ".ashx"
    or FileName endswith ".asmx"
| where FolderPath has_any (
    "inetpub", "wwwroot", "FrontEnd", "HttpProxy",
    "OWA", "ECP", "Autodiscover", "aspnet_client")
| where ActionType == "FileCreated"
| project Timestamp, DeviceName, FileName, FolderPath,
    SHA256, InitiatingProcessFileName
| sort by Timestamp desc
```

Hunt Procedure: Web Server Process Spawning (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect w3wp.exe spawning cmd/powershell (web shell activity)
DeviceProcessEvents
| where InitiatingProcessFileName in~ ("w3wp.exe", "httpd.exe")
| where FileName in~ ("cmd.exe", "powershell.exe", "net.exe",
    "whoami.exe", "ipconfig.exe", "systeminfo.exe",
    "certutil.exe", "nslookup.exe")
| project Timestamp, DeviceName, InitiatingProcessFileName,
    FileName, ProcessCommandLine, AccountName
| sort by Timestamp desc
```

Hunt Procedure: Web Shell Detection via SPL

DETECTION ONLY

SPL (SPLUNK)

```
index=windows sourcetype="WinEventLog:Sysmon" EventCode=1
| search ParentImage="*w3wp.exe"
    (Image="*cmd.exe" OR Image="*powershell.exe"
     OR Image="*net.exe" OR Image="*whoami.exe")
| table _time, ComputerName, User, ParentImage,
    Image, CommandLine
```

- ☐ Inventory all ASPX/ASHX files on Exchange and IIS servers
- ☐ Compare against known-good baselines from fresh installations
- ☐ Check for files created outside of patching or deployment windows
- ☐ Inspect file contents for encoded commands, eval(), or base64 strings

Escalation Criteria
Any new ASPX file on an Exchange server that was not part of a documented patch or deployment should be treated as a potential web shell. Collect the file for analysis and check its hash against known web shell signatures.

7 Dormant Implant Discovery

Duration: 35 min Priority: HIGH

Objective: Identify dormant APT34 implants using beaconing analysis, low-and-slow C2 patterns, and detection of pre-positioned access mechanisms that may activate during geopolitical escalation.

Background

APT34 is known for maintaining long-duration persistent access in victim networks. Between November 2024 and April 2025, Hunt.io tracked APT34-like infrastructure that remained dormant from a payload perspective, suggesting the group pre-positions access for future use. Following geopolitical events (Operation Epic Fury, February 2026), dormant implants are at elevated risk of activation. This module focuses on discovering access mechanisms that are already present but not actively communicating.

Hunt Procedure: Beaconing Pattern Analysis (KQL)

INVESTIGATION AID

KQL (MICROSOFT SENTINEL)

```
// Detect consistent outbound connection intervals (beaconing)
// APT34 implants check in at regular intervals via DNS or HTTP
DeviceNetworkEvents
| where RemotePort in (53, 80, 443)
| where ActionType == "ConnectionSuccess"
| order by DeviceName, RemoteUrl, Timestamp asc
| serialize
| extend PrevTimestamp = prev(Timestamp)
| extend Interval = datetime_diff("second", Timestamp, PrevTimestamp)
| summarize
    ConnectionCount = count(),
    AvgInterval = avg(Interval)
    by DeviceName, RemoteUrl, bin(Timestamp, 24h)
| where ConnectionCount > 20
    and AvgInterval between (50 .. 3700)  // 1-60 min intervals
| sort by ConnectionCount desc
```

Hunt Procedure: ngrok Reverse Tunnel Detection (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// APT34 uses ngrok for persistent reverse tunneling
// Source: Trend Micro Earth Simnavaz (Oct 2024)
DeviceProcessEvents
| where FileName =~ "ngrok.exe"
    or ProcessCommandLine has "ngrok"
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, AccountName,
    InitiatingProcessFileName
| sort by Timestamp desc
// Also check for ngrok DNS connections:
// DeviceNetworkEvents
// | where RemoteUrl has "ngrok" or RemoteUrl has "tunnel.us"
```

Hunt Procedure: Dormant Service Detection (KQL)

INVESTIGATION AID

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Find Windows services that were installed but rarely execute
// APT34 creates services (T1543.003) for persistence
DeviceRegistryEvents
| where RegistryKey has @"SYSTEM\CurrentControlSet\Services"
| where RegistryValueName == "ImagePath"
| where ActionType == "RegistryValueSet"
| where RegistryValueData !has_any (
    "svchost.exe", "System32", "SysWOW64",
    "Microsoft", "Windows")
| project Timestamp, DeviceName, RegistryKey,
    RegistryValueData, InitiatingProcessFileName
```

- ☐ Run beaconing analysis across all network logs for the past 90 days
- ☐ Check for ngrok installations or connections on all servers
- ☐ Audit Windows services for non-standard binary paths
- ☐ Review startup items and Run/RunOnce registry keys for unusual entries
- ☐ Check for Outlook Home Page rules (T1137.004), an APT34 persistence mechanism

Geopolitical Context
Following the February 2026 Operation Epic Fury strikes, Iranian APT groups are assessed at elevated operational tempo. Pre-positioned APT34 implants that have been dormant for months or years may activate. Organizations in government, energy, financial services, and telecommunications should treat dormant implant discovery as an immediate priority.

8 Lateral Movement via Pass-the-Hash/WMI

Duration: 25 min Priority: HIGH

Objective: Detect lateral movement patterns consistent with APT34's use of WMI (T1047), PsExec (S0029), RDP (T1021.001), and SSH (T1021.004) for moving through victim networks after credential harvesting.

Background

After harvesting credentials via Mimikatz or LaZagne, APT34 moves laterally using WMI, PsExec, and RDP. The Intruvent Codex confirms T1047 (WMI), T1021.001 (RDP), and T1021.004 (SSH) mappings for OilRig. The group has also been observed using valid domain accounts (T1078.002) for lateral movement, making detection dependent on behavioral analysis rather than tool signatures.

Hunt Procedure: WMI Remote Execution (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect WMI-based remote command execution
// APT34 uses WMI for lateral movement (MITRE T1047)
DeviceProcessEvents
| where FileName in~ ("wmic.exe", "wmiprvse.exe")
| where ProcessCommandLine has_any (
    "process call create", "node:", "/node:",
    "shadowcopy", "os get")
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, AccountName,
    InitiatingProcessFileName
| sort by Timestamp desc
```

Hunt Procedure: PsExec Lateral Movement (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect PsExec usage for remote execution
// APT34 uses PsExec (S0029) for lateral movement
DeviceProcessEvents
| where FileName in~ ("psexec.exe", "psexec64.exe", "PSEXESVC.exe")
    or ProcessCommandLine has "psexec"
| project Timestamp, DeviceName, FileName,
    ProcessCommandLine, AccountName,
    InitiatingProcessFileName
| sort by Timestamp desc
```

Hunt Procedure: Anomalous RDP Sessions (KQL)

INVESTIGATION AID

KQL (MICROSOFT SENTINEL)

```
// Identify RDP sessions from unusual source hosts
SecurityEvent
| where EventID == 4624
| where LogonType == 10  // RemoteInteractive (RDP)
| where IpAddress != "" and IpAddress != "-"
| summarize
    SessionCount = count(),
    SourceHosts = make_set(IpAddress, 10)
    by TargetUserName, Computer, bin(TimeGenerated, 24h)
| where SessionCount > 5
| sort by SessionCount desc
```

- ☐ Review WMI process creation events across all servers for the past 30 days
- ☐ Check for PsExec service installation events (PSEXESVC) on remote hosts
- ☐ Correlate RDP sessions with credential harvesting events from Module 4
- ☐ Look for SSH connections (T1021.004) from unexpected internal hosts

False Positive Guidance
WMI and PsExec are legitimate administrative tools used by IT operations teams. Focus on executions from non-admin workstations, executions outside business hours, and lateral movement patterns that follow credential harvesting events. SCCM and SCOM may trigger WMI alerts.

9 Data Staging and Exfiltration via DNS

Duration: 30 min Priority: HIGH

Objective: Detect data staging and exfiltration patterns used by APT34, including DNS-based data exfiltration, steganographic channels (RDAT BMP images), and automated collection mechanisms.

Background

APT34 exfiltrates data through multiple channels. DNS tunneling carries data encoded in subdomain labels (T1048.003). RDAT embeds data in BMP images attached to emails via Exchange Web Services. The group uses automated collection (T1119) and screen capture (T1113) to gather intelligence. Data is often staged before exfiltration using clipboard collection (T1115) and local file collection (T1005).

Hunt Procedure: DNS Exfiltration Volume Analysis (KQL)

DETECTION ONLY

KQL (MICROSOFT SENTINEL)

```
// Detect high-volume DNS queries that could indicate data exfiltration
// APT34 encodes stolen data in DNS subdomain labels
DnsEvents
| where TimeGenerated > ago(7d)
| extend DomainParts = split(Name, ".")
| extend BaseDomain = strcat(tostring(DomainParts[-2]), ".",
    tostring(DomainParts[-1]))
| summarize
    TotalQueries = count(),
    TotalDataBytes = sum(strlen(tostring(DomainParts[0]))),
    UniqueSubdomains = dcount(Name)
    by ClientIP, BaseDomain
| where TotalQueries > 500
    and UniqueSubdomains > 200
    and TotalDataBytes > 10000
| sort by TotalDataBytes desc
```

Hunt Procedure: Steganographic Exfiltration via Email (KQL)

INVESTIGATION AID

KQL (MICROSOFT SENTINEL)

```
// Detect BMP image attachments sent via Exchange (RDAT pattern)
// RDAT embeds data in BMP images attached to emails
OfficeActivity
| where Operation == "Send"
| where AttachmentCount > 0
| extend AttachmentExt = tostring(split(
    tostring(parse_json(tostring(AttachmentNames))[0]),
    ".")[-1])
| where AttachmentExt in~ ("bmp", "png", "jpg")
| where SenderIPAddress != ""
| summarize
    ImageEmailCount = count()
    by UserId, SenderIPAddress, bin(TimeGenerated, 24h)
| where ImageEmailCount > 5
| sort by ImageEmailCount desc
```

Hunt Procedure: Data Staging via SPL

DETECTION ONLY

SPL (SPLUNK)

```
`dns`
| eval query_domain=mvindex(split(query,"."), -2) . "." . mvindex(split(query,"."), -1)
| stats sum(len(query)) as total_bytes, count as query_count,
    dc(query) as unique_queries by src_ip, query_domain
| where total_bytes > 10000 AND unique_queries > 200
| sort -total_bytes
```

- ☐ Analyze DNS query logs for endpoints transmitting >10KB of data via subdomain labels in 24 hours
- ☐ Check Exchange message tracking logs for unusual image attachments from service accounts
- ☐ Review file creation events in temporary directories for data staging patterns
- ☐ Look for certutil or bitsadmin usage for file downloads (APT34 tools per MITRE G0049)

False Positive Guidance
DKIM/DMARC verification, legitimate email services, and CDN analytics platforms generate high-volume DNS queries. Focus on DNS traffic to domains that are not in your organization's known-good domain list. Image attachments in email are common; focus on patterns involving automated sending from compromised or service accounts.

10 Email Account Compromise and Mailbox Export Detection

Duration: 30 min Priority: CRITICAL

Objective: Detect APT34's use of Exchange Web Services for email-based C2 and exfiltration, Outlook Home Page persistence (T1137.004), and bulk mailbox export activity consistent with PowerExchange, MrPerfectionManager, and STEALHOOK operations.

Background

APT34 has increasingly adopted email-based C2 protocols. PowerExchange monitors mailbox inboxes for commands encoded in email subjects. MrPerfectionManager and STEALHOOK exfiltrate data as email attachments through compromised Exchange servers. The Outlook Home Page technique (T1137.004, confirmed in Codex for OilRig) enables persistent code execution by modifying the default page loaded when a user opens an Outlook folder. This allows malware execution every time the compromised user opens Outlook, without modifying macros or startup items.

Hunt Procedure: Outlook Home Page Modification (KQL)

DETECTION ONLY

KQL (MICROSOFT DEFENDER FOR ENDPOINT)

```
// Detect Outlook Home Page URL modifications (T1137.004)
// APT34 uses this technique for persistent code execution
DeviceRegistryEvents
| where RegistryKey has @"Software\Microsoft\Office"
    and RegistryKey has "Outlook"
    and RegistryKey has "WebView"
| where RegistryValueName has "URL"
| where ActionType == "RegistryValueSet"
| project Timestamp, DeviceName, RegistryKey,
    RegistryValueName, RegistryValueData,
    InitiatingProcessFileName
| sort by Timestamp desc
```

Hunt Procedure: Exchange EWS Anomalous Access (KQL)

DETECTION ONLY

KQL (MICROSOFT SENTINEL)

```
// Detect unusual EWS API access patterns
// PowerExchange and STEALHOOK use EWS for C2 and exfiltration
OfficeActivity
| where OfficeWorkload == "Exchange"
| where Operation in ("MailItemsAccessed", "Send", "MoveToDeletedItems")
| where ClientInfoString has "EWS"
| summarize
    OperationCount = count(),
    Operations = make_set(Operation, 10)
    by UserId, ClientIP, bin(TimeGenerated, 1h)
| where OperationCount > 50
| sort by OperationCount desc
```

Hunt Procedure: Bulk Mailbox Export Detection (KQL)

DETECTION ONLY

KQL (MICROSOFT SENTINEL)

```
// Detect bulk mailbox export using New-MailboxExportRequest
// APT34 exports mailboxes for intelligence collection
OfficeActivity
| where OfficeWorkload == "Exchange"
| where Operation in (
    "New-MailboxExportRequest",
    "New-ComplianceSearchAction",
    "Set-Mailbox")
| project TimeGenerated, UserId, Operation,
    Parameters, ClientIP
| sort by TimeGenerated desc
```

Hunt Procedure: Exchange Exfiltration via SPL

DETECTION ONLY

SPL (SPLUNK)

```
index=o365 sourcetype="o365:management:activity"
    Workload=Exchange
| search (Operation="MailItemsAccessed"
    OR Operation="Send")
| where like(ClientInfoString, "%EWS%")
| stats count by UserId, ClientIP, Operation
| where count > 50
| sort -count
```

- ☐ Check all mailboxes for Outlook Home Page URL modifications
- ☐ Review EWS access logs for automated patterns (high-volume, regular intervals)
- ☐ Look for mailbox export requests not associated with HR, legal, or compliance actions
- ☐ Check for emails sent with attachments from service accounts or recently compromised accounts
- ☐ Review Exchange transport rules for new forwarding rules to external addresses

Escalation Criteria
Any Outlook Home Page URL modification is high priority. Any bulk mailbox export request not associated with a documented HR/legal investigation warrants immediate review. EWS access from non-standard client IPs (outside corporate ranges) with high operation counts should trigger an account compromise investigation.

A Appendix A: Indicators of Compromise

Blocking Guidance: 3-Tier IOC Classification

Tier 1: Block (High Confidence)
All malware SHA256 hashes listed below: Confirmed APT34 tools with no legitimate use case. Safe for hash-based blocking and EDR prevention policies.

Tier 2: Alert / Monitor (Shared Infrastructure)
ngrok tunneling: ngrok is a legitimate development tool used by millions. Blocking ngrok entirely will disrupt legitimate workflows. Deploy as detection/alerting query. Require analyst review before action. Do NOT block at network level.

Microsoft 365 / Cloud service C2: APT34 cloud-based downloaders use legitimate Microsoft services. Blocking these services is not feasible. Monitor for anomalous API call patterns.

DNS tunneling domains: APT34 rotates C2 domains frequently. Rather than blocking individual domains, deploy DNS analytics to detect tunneling behavior patterns.

Tier 3: Investigate (Unconfirmed)
Pre-positioned infrastructure: Domains identified by Hunt.io impersonating Iraqi/UK entities were dormant at discovery. Investigate contextually based on your organization's connections to Middle Eastern or UK academic/technology entities.

File Indicators

Indicator	Type	Context	Source	Confidence
`47d3e6c389cfdbc9cf7eb61f3051c9f4e50e30cf2d97499144e023ae87d68d5a`	SHA256	SideTwist backdoor	NSFOCUS (2023); Check Point (2021)	High
`13c27e5049a7fc5a36416f2c1ae49c12438d45ce50a82a96d3f792bfdacf3dcd`	SHA256	Malicious document delivering SideTwist variant	NSFOCUS (2023)	High
`704360d765f6f1ef735594c7ff5fb6c47467dad8abc3133f8e935a6c0c804c8a`	SHA256	APT34/OilRig malware sample	Joe Sandbox (2024)	High

Additional IOCs Available from Original Reports
APT34/OilRig has a malware arsenal spanning over a decade. Full IOC sets should be obtained directly from:

- **Palo Alto Unit 42:** Comprehensive OilRig Playbook (2017-2023) with Helminth, ISMAgent, BONDUPDATER, QUADAGENT hashes
- **ESET:** Solar, Mango, SampleCheck5000, OilCheck, OilBooster, ODAgent hashes (2023)
- **Trend Micro:** STEALHOOK and CVE-2024-30088 exploit hashes (October 2024)
- **Check Point:** SideTwist and Iraq campaign malware hashes (2021, 2024)
- **Malwarebytes/Fortinet:** Saitama backdoor hashes (2022)

PDB Paths

Indicator	Context	Source
`E:\Saitama\Saitama.Agent\obj\Release\Saitama.Agent.pdb`	Saitama backdoor (.NET), DNS tunneling C2	Malwarebytes (2022); Fortinet (2022)

CVEs

CVE	Product	Context	Source
`CVE-2024-30088`	Windows Kernel	TOCTOU race condition exploited for privilege escalation in Earth Simnavaz UAE campaign. Patched June 2024.	Trend Micro (Oct 2024)

B Appendix B: Detection Rules Summary

Detection Query Index

Module	Query Name	Platform	Safety Label	MITRE Technique
1	High-Entropy Subdomain Detection	KQL	DETECTION ONLY	T1071.004
1	Excessive TXT Record Queries	KQL	DETECTION ONLY	T1071.004
1	DNS Query Volume Anomaly	SPL	DETECTION ONLY	T1071.004
2	Known OilRig File Hashes	KQL	BLOCKING SAFE	Multiple
2	Saitama PDB Path Detection	KQL	BLOCKING SAFE	T1059
2	OilRig Behavioral Indicators	KQL	DETECTION ONLY	T1059.003
3	Password Filter DLL Registration	KQL	DETECTION ONLY	T1556.002
3	RunPE-In-Memory Indicators	KQL	DETECTION ONLY	T1068
3	CVE-2024-30088 Patch Verification	PowerShell	INVESTIGATION AID	T1068
4	LSASS Access Detection	KQL	DETECTION ONLY	T1003.001
4	LaZagne Execution Detection	KQL	DETECTION ONLY	T1555
4	Credential Dump Detection	SPL	DETECTION ONLY	T1003.001
5	Suspicious Scheduled Task Creation	KQL	DETECTION ONLY	T1053.005
5	Scheduled Task Audit	SPL	DETECTION ONLY	T1053.005
6	New ASPX/ASHX Files on Web Servers	KQL	DETECTION ONLY	T1505.003
6	Web Server Process Spawning	KQL	DETECTION ONLY	T1505.003
6	Web Shell Detection	SPL	DETECTION ONLY	T1505.003
7	Beaconing Pattern Analysis	KQL	INVESTIGATION AID	T1071.004
7	ngrok Reverse Tunnel Detection	KQL	DETECTION ONLY	T1572
7	Dormant Service Detection	KQL	INVESTIGATION AID	T1543.003
8	WMI Remote Execution	KQL	DETECTION ONLY	T1047
8	PsExec Lateral Movement	KQL	DETECTION ONLY	T1021
8	Anomalous RDP Sessions	KQL	INVESTIGATION AID	T1021.001
9	DNS Exfiltration Volume Analysis	KQL	DETECTION ONLY	T1048.003
9	Steganographic Exfiltration via Email	KQL	INVESTIGATION AID	T1048.003
9	Data Staging Detection	SPL	DETECTION ONLY	T1048.003
10	Outlook Home Page Modification	KQL	DETECTION ONLY	T1137.004
10	Exchange EWS Anomalous Access	KQL	DETECTION ONLY	T1071.001
10	Bulk Mailbox Export Detection	KQL	DETECTION ONLY	T1005
10	Exchange Exfiltration Detection	SPL	DETECTION ONLY	T1071.001

C Appendix C: References

Government Advisories & Standards Bodies

[1] **MITRE ATT&CK** - "Group G0049: OilRig"
attack.mitre.org

[2] **CISA** - "Iranian Government-Sponsored APT Actors Exploiting Microsoft Exchange and Fortinet Vulnerabilities" (AA21-321A)
cisa.gov

Vendor Threat Intelligence (Primary)

[3] **Palo Alto Unit 42** - "DNS Tunneling in the Wild: Overview of OilRig's DNS Tunneling" (2019)
unit42.paloaltonetworks.com

[4] **Palo Alto Unit 42** - "OilRig Deploys ALMA Communicator: DNS Tunneling Trojan" (2017)
unit42.paloaltonetworks.com

[5] **Trend Micro** - "Earth Simnavaz (aka APT34) Levies Advanced Cyberattacks Against Middle East" (October 2024)
trendmicro.com

[6] **ESET Research** - "OilRig's persistent attacks using cloud service-powered downloaders" (2023)
welivesecurity.com

[7] **Check Point Research** - "Iran's APT34 Returns with an Updated Arsenal" (April 2021)
research.checkpoint.com

[8] **Symantec (Broadcom)** - "Crambus: New Toolset Targeting Middle Eastern Government" (October 2023)
symantec-enterprise-blogs.security.com

[9] **NSFOCUS** - "APT34 Unleashes New Wave of Phishing Attack with Variant of SideTwist Trojan" (2023)
nsfocusglobal.com

[10] **FireEye (Mandiant)** - "New Targeted Attack in the Middle East by APT34" (December 2017)
mandiant.com

[11] **Hunt.io** - "APT34-Like Threat Infrastructure Uncovered Before Activation" (2025)
hunt.io

Secondary Sources

[12] **Malwarebytes** - "APT34 targets Jordan Government using new Saitama backdoor" (May 2022)
threatdown.com

[13] **The Hacker News** - "OilRig Exploits Windows Kernel Flaw in Espionage Campaign Targeting UAE and Gulf" (October 2024)
thehackernews.com

[14] **SOC Prime** - "Earth Simnavaz (aka APT34) Attack Detection" (2024)
socprime.com

[15] **Brandefense** - "OilRig: Iran's Persistent Espionage Arm In Cyberspace" (2025)
brandefense.io

[16] **Cyble** - "OilRig: Threat Actor Profile"
cyble.com

[17] **Picus Security** - "OilRig Exposed: Unveiling the Tools and Techniques of APT34"
picussecurity.com

[18] **Security Affairs** - "Analyzing OilRig's malware that uses DNS Tunneling"
securityaffairs.com

Data Sources

[19] **Intruvent Golden CTI Database (Codex)** - Actor profile: OilRig (G0049). 75 techniques, 30 malware/tool associations, 2 campaigns (Outer Space C0042, Juicy Mix C0044). Queried March 18, 2026.

NEED HELP IMPLEMENTING THESE DETECTIONS?

Our threat hunting team can deploy, tune, and operationalize these detection rules in your environment.

- ✓ Detection rule deployment and tuning
- ✓ Managed threat hunting operations
- ✓ Incident response retainer

Contact Our Team

or

Automate continuous detection with BRACE, our sector-specific threat intelligence platform. See BRACE plans →