THREAT INTELLIGENCE

Lotus Blossom / Chrysalis Backdoor Splunk SIEM Detection Rules

Comprehensive detection package for the Lotus Blossom APT campaign targeting the Chrysalis backdoor toolkit and associated infrastructure.

Reference: Rapid7 - Chrysalis Backdoor: Dive into Lotus Blossom's Toolkit
IOC Source: Neo23x0/signature-base (filename-iocs.txt) + Rapid7 campaign report
Date: February 2026
Classification: CONFIDENTIAL - SOC USE ONLY
Total Detection Rules: 39 searches across host and network IOCs
MITRE ATT&CK Coverage: 15+ techniques across 6 tactics

IOC SUMMARY

C2 IPs (9): 45.76.155.202, 45.77.31.210, 59.110.7.32, 61.4.102.97, 95.179.213.0,
124.222.137.114, 45.32.144.255, 160.250.93.48, 103.159.133.178
C2 Domains (7): api.skycloudcenter.com, api.wiresguard.com, cdncheck.it.com,
temp.sh, self-dns.it.com, safe-dns.it.com, cloudtrafficservice.com
C2 Ports: 8880 (59.110.7.32 - Alibaba), 9999 (124.222.137.114 - Tencent)
SHA256 Hashes (5): Chrysalis DLL Loader, Chrysalis Backdoor, CobaltStrike Loaders, Warbird Loader
Suspicious User-Agents: Chrome/80.0.4044.92, Chrome/92.0.4472.114, Chrome/134.0.0.0
GUP.exe: Notepad++ updater abuse (supply chain / DLL side-loading vector)

TABLE OF CONTENTS

Search 36a — INTRUSION DETECTION: ZenArmor/Suricata IDS alerts for C2 traffic

Search 36b — INTRUSION DETECTION: CIM Intrusion Detection Data Model

Search 37 — NETWORK IOC LOOKUP TABLE: CSV-based approach for production

Search 38 — COMBINED NETWORK + HOST ALERT: Full campaign detection (for Splunk Alert)

Search 39 — EXECUTIVE NETWORK IOC DASHBOARD: Campaign network activity overview

PART A — Host-Based Detection (Endpoint IOCs)

Search 1 — MASTER DETECTION: All Lotusblossom IOCs across Sysmon + Windows Events

Primary all-in-one search covering all IOCs across Sysmon data. Sysmon EventCodes: 1=ProcessCreate, 6=DriverLoad, 7=ImageLoad, 11=FileCreate, 15=FileCreateStreamHash, 23=FileDelete, 26=FileDeleteDetected.

Severity: CRITICAL / HIGH / MEDIUM | **MITRE:** TA0002, TA0003, TA0005, TA0009, TA0011 | T1036, T1547, T1074, T1574, T1055, T1132, T1105, T1027, T1059 | **Data Sources:** Sysmon (EventCode 1, 6, 7, 11, 15, 23, 26)

SPL — Search 1

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode IN (1, 6, 7, 11, 15, 23, 26)
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "libtcc.dll"
    )
| eval target_path=coalesce(TargetFilename, Image, ImageLoaded, TargetObject)
| where isnotnull(target_path)
| eval ioc_match=case(
    match(target_path, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$"), "USOShared Suspicious Binary/Source",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\[a-zA-Z0-9]{1}\.txt$"), "ProShow Single-Char TXT (Staging)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\[a-zA-Z0-9]{1}\.txt$"), "Adobe Scripts Single-Char TXT (Staging)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$"), "Bluetooth log.dll (Chrysalis Loader)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$"), "Fake BluetoothService.exe (Chrysalis)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$"), "ProShow 'load' (No Extension - Payload)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$"), "Fake ProShow.exe (Chrysalis Persistence)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$"), "Adobe Scripts alien.ini (Config)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$"), "Adobe Scripts script.exe (Chrysalis)",
    match(target_path, "(?i)\\\\libtcc\.dll$"), "libtcc.dll (TinyCC Compiler - Suspicious)",
    1==1, null()
)
| where isnotnull(ioc_match)
| eval severity=case(
    match(ioc_match, "Chrysalis|Loader|Payload"), "Critical",
    match(ioc_match, "USOShared|Staging|Config"), "High",
    match(ioc_match, "libtcc"), "Medium",
    1==1, "High"
)
| eval mitre_tactics=case(
    match(ioc_match, "Persistence|Fake"), "TA0003 - Persistence",
    match(ioc_match, "Staging|Single-Char"), "TA0005 - Defense Evasion, TA0009 - Collection",
    match(ioc_match, "Loader|Payload"), "TA0002 - Execution",
    match(ioc_match, "Config"), "TA0011 - Command and Control",
    match(ioc_match, "libtcc"), "TA0002 - Execution, TA0005 - Defense Evasion",
    1==1, "TA0002 - Execution"
)
| eval mitre_techniques=case(
    match(ioc_match, "Persistence|Fake|ProShow.exe|BluetoothService"), "T1036 - Masquerading, T1547 - Boot or Logon Autostart",
    match(ioc_match, "Staging|Single-Char"), "T1074 - Data Staged, T1036 - Masquerading",
    match(ioc_match, "Loader|log.dll"), "T1574 - Hijack Execution Flow, T1055 - Process Injection",
    match(ioc_match, "Config|alien.ini"), "T1132 - Data Encoding, T1105 - Ingress Tool Transfer",
    match(ioc_match, "libtcc"), "T1027 - Obfuscated Files, T1059 - Command and Scripting",
    match(ioc_match, "USOShared"), "T1036 - Masquerading, T1574 - Hijack Execution Flow",
    1==1, "T1036 - Masquerading"
)
| table _time, host, EventCode, ioc_match, severity, mitre_tactics, mitre_techniques, target_path, User, ProcessId, ParentImage, ParentCommandLine,
| sort - _time
```

Search 2 — SYSMON FILE CREATION: Lotusblossom file drops (EventCode 11)

Detects file creation events matching Lotusblossom staging/implant patterns. Often the first indicator of compromise during initial access.

Severity: CRITICAL / HIGH | **MITRE:** T1074 - Data Staged, T1036 - Masquerading | **Data Sources:** Sysmon (EventCode 11 - FileCreate)

SPL — Search 2

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=11
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "libtcc.dll"
    )
| eval target=TargetFilename
| where match(target, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\[a-zA-Z0-9]{1}\.txt$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\[a-zA-Z0-9]{1}\.txt$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$")
    OR match(target, "(?i)\\\\libtcc\.dll$")
| eval ioc_match=case(
    match(target, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$"), "USOShared Suspicious File Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\[a-zA-Z0-9]{1}\.txt$"), "ProShow Single-Char TXT Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\[a-zA-Z0-9]{1}\.txt$"), "Adobe Scripts Single-Char TXT Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$"), "Chrysalis Loader DLL Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$"), "Fake BluetoothService Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$"), "ProShow Payload Drop (No Extension)",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$"), "Fake ProShow.exe Drop",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$"), "Chrysalis Config Drop (alien.ini)",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$"), "Chrysalis Backdoor Drop (script.exe)",
    match(target, "(?i)\\\\libtcc\.dll$"), "TinyCC Compiler DLL Drop",
    1==1, "Unknown Lotusblossom IOC"
)
| table _time, host, User, Image, target, ioc_match, Hashes, ProcessId
| sort - _time
```

Search 3 — SYSMON PROCESS CREATION: Lotusblossom executable launches (EventCode 1)

Detects process creation events from known Lotusblossom implant locations. These indicate active execution of the backdoor or its components.

Severity: CRITICAL | **MITRE:** T1036 - Masquerading, T1059 - Command and Scripting Interpreter | **Data Sources:** Sysmon (EventCode 1 - ProcessCreate)

SPL — Search 3

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=1
    (
        ("USOShared" AND ("*.exe" OR "*.dll"))
        OR ("ProShow" AND "ProShow.exe")
        OR ("Adobe" AND "Scripts" AND "script.exe")
        OR ("Bluetooth" AND "BluetoothService.exe")
    )
| eval proc=Image
| where match(proc, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$")
| eval ioc_match=case(
    match(proc, "(?i)\\\\USOShared\\\\"), "CRITICAL: USOShared Executable Launch",
    match(proc, "(?i)\\\\ProShow\\\\ProShow\.exe$"), "CRITICAL: Fake ProShow.exe Execution",
    match(proc, "(?i)\\\\Adobe\\\\Scripts\\\\script\.exe$"), "CRITICAL: Chrysalis Backdoor Execution",
    match(proc, "(?i)\\\\Bluetooth\\\\BluetoothService\.exe$"), "CRITICAL: Fake BluetoothService Execution",
    1==1, "Lotusblossom Process Execution"
)
| table _time, host, User, ioc_match, Image, CommandLine, ParentImage, ParentCommandLine, Hashes, ProcessId, ParentProcessId
| sort - _time
```

Search 4 — SYSMON IMAGE/DLL LOAD: Lotusblossom DLL side-loading (EventCode 7)

Detects DLL loading events matching Lotusblossom implant patterns. DLL side-loading is a key technique used by Lotus Blossom for persistence.

Severity: CRITICAL / HIGH | **MITRE:** T1574.002 - DLL Side-Loading | **Data Sources:** Sysmon (EventCode 7 - ImageLoad)

SPL — Search 4

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=7
    (
        ("USOShared" AND "*.dll")
        OR ("Bluetooth" AND "log.dll")
        OR "libtcc.dll"
    )
| eval loaded_dll=ImageLoaded
| where match(loaded_dll, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.dll$")
    OR match(loaded_dll, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$")
    OR match(loaded_dll, "(?i)\\\\libtcc\.dll$")
| eval ioc_match=case(
    match(loaded_dll, "(?i)\\\\USOShared\\\\"), "USOShared Suspicious DLL Load",
    match(loaded_dll, "(?i)\\\\Bluetooth\\\\log\.dll$"), "CRITICAL: Chrysalis Loader DLL Loaded",
    match(loaded_dll, "(?i)\\\\libtcc\.dll$"), "Suspicious libtcc.dll Load (TinyCC)",
    1==1, "Lotusblossom DLL Load"
)
| table _time, host, User, ioc_match, Image, loaded_dll, Hashes, Signed, Signature, SignatureStatus, ProcessId
| sort - _time
```

Search 5 — WINDOWS SECURITY LOGS: File access/creation audit events

Detects Windows Security audit events (4663, 4656, 4660) for Lotusblossom paths. Requires Object Access auditing enabled on target systems.

Severity: HIGH | **MITRE:** T1083 - File and Directory Discovery | **Data Sources:** Windows Security (EventCode 4663, 4656, 4660)

SPL — Search 5

```
index=* sourcetype IN ("WinEventLog:Security", "wineventlog:security", "XmlWinEventLog:Security", "xmlwineventlog:security")
    EventCode IN (4663, 4656, 4660)
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "libtcc.dll"
    )
| eval obj=coalesce(ObjectName, Object_Name, ObjectName)
| where match(obj, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$")
    OR match(obj, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\")
    OR match(obj, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\")
    OR match(obj, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\(log\.dll|BluetoothService\.exe)$")
    OR match(obj, "(?i)\\\\libtcc\.dll$")
| eval ioc_match=case(
    match(obj, "(?i)\\\\USOShared\\\\"), "USOShared File Access",
    match(obj, "(?i)\\\\ProShow\\\\"), "ProShow Directory Access",
    match(obj, "(?i)\\\\Adobe\\\\Scripts\\\\"), "Adobe Scripts Directory Access",
    match(obj, "(?i)\\\\Bluetooth\\\\log\.dll$"), "Chrysalis Loader Access",
    match(obj, "(?i)\\\\Bluetooth\\\\BluetoothService\.exe$"), "Fake BluetoothService Access",
    match(obj, "(?i)\\\\libtcc\.dll$"), "libtcc.dll Access",
    1==1, "Lotusblossom File Access"
)
| table _time, host, Account_Name, ioc_match, EventCode, obj, AccessMask, ProcessName, ProcessId
| sort - _time
```

Search 6 — CIM ENDPOINT DATA MODEL: Filesystem Changes (CIM-compliant)

Uses Splunk CIM Endpoint.Filesystem data model for normalized detection. Works across any CIM-compliant data source (Sysmon, EDR, etc.).

Severity: CRITICAL / HIGH | **MITRE:** T1074 - Data Staged, T1036 - Masquerading | **Data Sources:** CIM Endpoint.Filesystem Data Model

SPL — Search 6

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(Filesystem.user) as user, values(Filesystem.p
    FROM datamodel=Endpoint.Filesystem
    WHERE (
        Filesystem.file_path="*USOShared*"
        OR Filesystem.file_path="*ProShow*"
        OR (Filesystem.file_path="*Adobe*Scripts*")
        OR (Filesystem.file_path="*Bluetooth*log.dll*")
        OR (Filesystem.file_path="*Bluetooth*BluetoothService*")
        OR Filesystem.file_path="*libtcc.dll*"
    )
    BY Filesystem.dest, Filesystem.file_path, Filesystem.file_name
| rename "Filesystem.*" as *
| eval ioc_match=case(
    match(file_path, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$"), "USOShared Suspicious File",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\[a-zA-Z0-9]{1}\.txt$"), "ProShow Single-Char TXT",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\[a-zA-Z0-9]{1}\.txt$"), "Adobe Scripts Single-Char TXT",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$"), "Chrysalis Loader (log.dll)",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$"), "Fake BluetoothService.exe",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$"), "ProShow Payload (no ext)",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$"), "Fake ProShow.exe",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$"), "Chrysalis Config (alien.ini)",
    match(file_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$"), "Chrysalis Backdoor (script.exe)",
    match(file_path, "(?i)\\\\libtcc\.dll$"), "libtcc.dll (TinyCC)",
    match(file_path, "(?i)\\\\USOShared\\\\"), "USOShared Directory Activity",
    match(file_path, "(?i)\\\\ProShow\\\\"), "ProShow Directory Activity",
    match(file_path, "(?i)\\\\Adobe\\\\Scripts\\\\"), "Adobe Scripts Directory Activity",
    1==1, null()
)
| where isnotnull(ioc_match)
| table first_seen, last_seen, dest, user, ioc_match, file_path, file_name, process, action, count
| sort - last_seen
```

Search 7 — CIM ENDPOINT DATA MODEL: Process Launches (CIM-compliant)

Uses Splunk CIM Endpoint.Processes data model for normalized process detection.

Severity: CRITICAL | **MITRE:** T1036 - Masquerading, T1059 - Command and Scripting | **Data Sources:** CIM Endpoint.Processes Data Model

SPL — Search 7

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(Processes.parent_process) as parent_process,
    FROM datamodel=Endpoint.Processes
    WHERE (
        Processes.process_path="*USOShared*"
        OR Processes.process_path="*ProShow*ProShow.exe*"
        OR Processes.process_path="*Adobe*Scripts*script.exe*"
        OR Processes.process_path="*Bluetooth*BluetoothService.exe*"
        OR Processes.process="*libtcc.dll*"
    )
    BY Processes.dest, Processes.process_name, Processes.process_path
| rename "Processes.*" as *
| eval ioc_match=case(
    match(process_path, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.exe$"), "CRITICAL: USOShared Process Execution",
    match(process_path, "(?i)\\\\ProShow\\\\ProShow\.exe$"), "CRITICAL: Fake ProShow.exe Execution",
    match(process_path, "(?i)\\\\Adobe\\\\Scripts\\\\script\.exe$"), "CRITICAL: Chrysalis Backdoor Execution",
    match(process_path, "(?i)\\\\Bluetooth\\\\BluetoothService\.exe$"), "CRITICAL: Fake BluetoothService Execution",
    match(process_cmdline, "(?i)libtcc\.dll"), "SUSPICIOUS: libtcc.dll Referenced in Process",
    1==1, null()
)
| where isnotnull(ioc_match)
| table first_seen, last_seen, dest, user, ioc_match, process_name, process_path, process_cmdline, parent_process, count
| sort - last_seen
```

Search 8 — DIRECTORY EXISTENCE CHECK: Suspicious Lotusblossom directories

Detects creation of or access to the specific directory structures used by Lotus Blossom for staging. These directories mimic legitimate software but are not standard Windows paths.

Severity: HIGH | **MITRE:** T1036 - Masquerading, T1074 - Data Staged | **Data Sources:** Sysmon (EventCode 1, 11, 12, 13)

SPL — Search 8

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode IN (1, 11, 12, 13)
    (
        ("AppData\\Roaming\\ProShow")
        OR ("AppData\\Roaming\\Adobe\\Scripts")
        OR ("AppData\\Roaming\\Bluetooth" AND NOT "Microsoft")
    )
| eval target=coalesce(TargetFilename, Image, TargetObject)
| where match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\")
| eval suspicious_dir=case(
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\"), "ProShow (Not standard Windows software)",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\"), "Adobe\\Scripts (Non-standard Adobe path)",
    match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\"), "Bluetooth (Impersonating system service)",
    1==1, "Unknown"
)
| stats count, earliest(_time) as first_seen, latest(_time) as last_seen, values(EventCode) as event_codes, values(target) as files_observed, dc(ta
    BY host, User, suspicious_dir
| where unique_files > 0
| sort - last_seen
```

Search 9 — USOSHARED ABUSE DETECTION: Files in Windows Update Orchestrator path

USOShared (Update Session Orchestrator) is a legitimate Windows directory but Lotus Blossom drops malicious .c, .dll, and .exe files there. Legitimate USOShared should only contain .etl (trace) and log files.

Severity: HIGH | **MITRE:** T1036 - Masquerading, T1574 - Hijack Execution Flow | **Data Sources:** Sysmon (EventCode 1, 7, 11, 15)

SPL — Search 9

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode IN (1, 7, 11, 15)
    "USOShared"
| eval target=coalesce(TargetFilename, Image, ImageLoaded)
| where match(target, "(?i)\\\\USOShared\\\\")
    AND NOT match(target, "(?i)\.(etl|log)$")
| eval is_known_ioc=if(match(target, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$"), "YES - Matches Lotusblossom IOC Pattern", "SUSPICIOUS
| table _time, host, User, EventCode, is_known_ioc, target, Image, Hashes, ProcessId
| sort - _time
```

Search 10 — WINDOWS DEFENDER / AV DETECTION: Correlation with AV alerts

Checks if Windows Defender or other AV has flagged any files in Lotusblossom paths.

Severity: HIGH | **MITRE:** T1562.001 - Disable or Modify Tools | **Data Sources:** Windows Defender Operational Log

SPL — Search 10

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Windows Defender/Operational", "WinEventLog:Microsoft-Windows-Windows Defender/Operational
    EventCode IN (1006, 1007, 1008, 1009, 1010, 1015, 1116, 1117, 1118, 1119)
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "libtcc"
        OR "Chrysalis"
        OR "LotusB"
    )
| table _time, host, EventCode, Threat_Name, Severity, Path, Action, User
| sort - _time
```

Search 11 — REGISTRY PERSISTENCE: Autorun entries pointing to Lotusblossom paths

Detects registry modifications (Sysmon EventCode 12/13/14) creating persistence mechanisms that point to known Lotusblossom implant locations.

Severity: CRITICAL | **MITRE:** T1547.001 - Registry Run Keys, T1112 - Modify Registry | **Data Sources:** Sysmon (EventCode 12, 13, 14)

SPL — Search 11

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode IN (12, 13, 14)
    (
        "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "USOShared"
        OR "libtcc"
    )
| eval reg_target=coalesce(TargetObject, "")
| eval reg_details=coalesce(Details, "")
| where (match(reg_target, "(?i)(Run|RunOnce|Winlogon|Shell|Userinit|Explorer|Services|Image File Execution)")
    AND (match(reg_details, "(?i)(ProShow|Adobe\\\\Scripts|Bluetooth\\\\(log\.dll|BluetoothService)|USOShared|libtcc)")))
    OR match(reg_details, "(?i)(ProShow\\\\ProShow\.exe|Adobe\\\\Scripts\\\\script\.exe|Bluetooth\\\\BluetoothService\.exe)")
| eval ioc_match="CRITICAL: Registry Persistence - Lotusblossom"
| table _time, host, User, EventCode, ioc_match, reg_target, reg_details, Image, ProcessId
| sort - _time
```

Search 12 — NETWORK CORRELATION: Processes from Lotusblossom paths making connections

Sysmon EventCode 3 (NetworkConnect) from processes in suspicious locations. Detects C2 communication from active Chrysalis implants.

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol, T1573 - Encrypted Channel | **Data Sources:** Sysmon (EventCode 3 - NetworkConnect)

SPL — Search 12

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=3
    (
        "USOShared"
        OR ("ProShow" AND "ProShow.exe")
        OR ("Adobe" AND "Scripts" AND "script.exe")
        OR ("Bluetooth" AND "BluetoothService.exe")
    )
| eval proc=Image
| where match(proc, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$")
    OR match(proc, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$")
| eval ioc_match="CRITICAL: Lotusblossom C2 Communication Detected"
| table _time, host, User, ioc_match, proc, DestinationIp, DestinationPort, DestinationHostname, SourceIp, SourcePort, Protocol
| sort - _time
```

Search 13 — SCHEDULED TASK / SERVICE PERSISTENCE

Detects scheduled tasks or services created that reference Lotusblossom paths. Windows Event 4698=Task Created, 4702=Task Updated, 7045=Service Installed.

Severity: CRITICAL | **MITRE:** T1053.005 - Scheduled Task, T1543.003 - Windows Service | **Data Sources:** Windows Security (4698, 4702), Windows System (7045)

SPL — Search 13

```
index=* sourcetype IN ("WinEventLog:Security", "wineventlog:security", "XmlWinEventLog:Security", "xmlwineventlog:security", "WinEventLog:System",
    EventCode IN (4698, 4702, 7045)
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" AND "Scripts")
        OR ("Bluetooth" AND ("log.dll" OR "BluetoothService"))
        OR "libtcc"
    )
| eval task_content=coalesce(TaskContent, Task_Content, ServiceFileName, Service_File_Name, ImagePath, "")
| where match(task_content, "(?i)(USOShared|ProShow|Adobe\\\\Scripts|Bluetooth\\\\(log\.dll|BluetoothService)|libtcc)")
| eval ioc_match="CRITICAL: Scheduled Task/Service Persistence - Lotusblossom"
| table _time, host, Account_Name, EventCode, ioc_match, TaskName, task_content, ServiceName, ServiceType
| sort - _time
```

Search 14 — COMPREHENSIVE STATS DASHBOARD: Summary view for SOC analysts

Overview of all Lotusblossom activity across the environment. Use for dashboarding and periodic threat hunting reviews.

Severity: INFO / HIGH | **MITRE:** Multiple | **Data Sources:** Sysmon, Windows Security, Windows System, Windows Defender

SPL — Search 14

```
index=*
    (
        "USOShared"
        OR "ProShow"
        OR ("Adobe" "Scripts" ("alien.ini" OR "script.exe" OR ".txt"))
        OR ("Bluetooth" ("log.dll" OR "BluetoothService"))
        OR "libtcc.dll"
    )
    sourcetype IN (
        "XmlWinEventLog:Microsoft-Windows-Sysmon/Operational",
        "xmlwineventlog:microsoft-windows-sysmon/operational",
        "WinEventLog:Security", "wineventlog:security",
        "XmlWinEventLog:Security", "xmlwineventlog:security",
        "WinEventLog:System", "wineventlog:system",
        "XmlWinEventLog:System", "xmlwineventlog:system",
        "WinEventLog:Microsoft-Windows-Windows Defender/Operational",
        "XmlWinEventLog:Microsoft-Windows-Windows Defender/Operational"
    )
| eval target_path=coalesce(TargetFilename, Image, ImageLoaded, TargetObject, ObjectName, Object_Name, "N/A")
| eval ioc_category=case(
    match(target_path, "(?i)\\\\USOShared\\\\"), "USOShared Abuse",
    match(target_path, "(?i)\\\\ProShow\\\\"), "ProShow Impersonation",
    match(target_path, "(?i)\\\\Adobe\\\\Scripts\\\\"), "Adobe Scripts Abuse",
    match(target_path, "(?i)\\\\Bluetooth\\\\(log\.dll|BluetoothService)"), "Bluetooth Impersonation",
    match(target_path, "(?i)libtcc\.dll"), "TinyCC Compiler DLL",
    1==1, "Other Lotusblossom Indicator"
)
| stats count as total_events, dc(host) as unique_hosts, earliest(_time) as first_seen, latest(_time) as last_seen, values(host) as affected_hosts,
    BY ioc_category
| eval first_seen=strftime(first_seen, "%Y-%m-%d %H:%M:%S")
| eval last_seen=strftime(last_seen, "%Y-%m-%d %H:%M:%S")
| sort - total_events
```

Search 15 — ALERT: High-Confidence Lotusblossom Detection (for Splunk Alerts)

Simplified, high-confidence search suitable for creating a Splunk alert. Focuses on the most specific IOCs with lowest false-positive rate. Recommended: Schedule every 5-15 minutes.

Severity: CRITICAL | **MITRE:** T1036 - Masquerading, T1574 - Hijack Execution Flow | **Data Sources:** Sysmon (EventCode 1, 7, 11)

SPL — Search 15

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode IN (1, 7, 11)
    (
        ("Bluetooth" AND ("log.dll" OR "BluetoothService.exe"))
        OR ("ProShow" AND ("ProShow.exe" OR "load"))
        OR ("Adobe" AND "Scripts" AND ("alien.ini" OR "script.exe"))
    )
| eval target=coalesce(TargetFilename, Image, ImageLoaded)
| where match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$")
    OR match(target, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$")
| eval alert_title="CRITICAL: Lotusblossom/Chrysalis Backdoor Activity Detected"
| eval ioc_detail=case(
    match(target, "(?i)log\.dll$"), "Chrysalis loader DLL (log.dll) in fake Bluetooth directory",
    match(target, "(?i)BluetoothService\.exe$"), "Chrysalis masquerading as BluetoothService.exe",
    match(target, "(?i)ProShow\.exe$"), "Chrysalis masquerading as ProShow.exe",
    match(target, "(?i)\\\\load$"), "Chrysalis payload 'load' (extensionless) in ProShow directory",
    match(target, "(?i)alien\.ini$"), "Chrysalis C2 configuration file (alien.ini)",
    match(target, "(?i)script\.exe$"), "Chrysalis backdoor executable (script.exe) in Adobe Scripts",
    1==1, "Lotusblossom IOC Match"
)
| eval response_action="IMMEDIATE: Isolate host, capture memory dump, preserve Sysmon/Event logs, begin IR investigation per Lotus Blossom playbook
| table _time, host, User, alert_title, ioc_detail, response_action, EventCode, target, Image, Hashes, CommandLine, ParentImage
| sort - _time
```

PART B — Network-Based Detection (C2 IOCs)

Search 25 — SYSMON NETWORK: C2 IP IOC Detection (EventCode 3)

Detects outbound connections to known Lotus Blossom C2 infrastructure via Sysmon. Sysmon EventCode 3 = NetworkConnect. Primary network IOC detection.

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol, T1573 - Encrypted Channel, T1105 - Ingress Tool Transfer | **Data Sources:** Sysmon (EventCode 3 - NetworkConnect)

SPL — Search 25

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=3
    (
        DestinationIp IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.2
        OR DestinationHostname IN ("api.skycloudcenter.com", "api.wiresguard.com", "cdncheck.it.com", "temp.sh", "self-dns.it.com", "safe-dns.it.co
    )
| eval ioc_match=case(
    DestinationIp=="45.76.155.202", "C2 IP: 45.76.155.202 (Vultr VPS - Lotus Blossom C2)",
    DestinationIp=="45.77.31.210", "C2 IP: 45.77.31.210 (Vultr VPS - Lotus Blossom C2)",
    DestinationIp=="59.110.7.32", "C2 IP: 59.110.7.32 (Alibaba Cloud - Chrysalis C2, port 8880)",
    DestinationIp=="61.4.102.97", "C2 IP: 61.4.102.97 (Lotus Blossom C2)",
    DestinationIp=="95.179.213.0", "C2 IP: 95.179.213.0 (Vultr VPS - Lotus Blossom C2)",
    DestinationIp=="124.222.137.114", "C2 IP: 124.222.137.114 (Tencent Cloud - Chrysalis C2, port 9999)",
    DestinationIp=="45.32.144.255", "C2 IP: 45.32.144.255 (Vultr VPS - Lotus Blossom C2)",
    DestinationIp=="160.250.93.48", "C2 IP: 160.250.93.48 (Lotus Blossom C2)",
    DestinationIp=="103.159.133.178", "C2 IP: 103.159.133.178 (Lotus Blossom C2)",
    match(DestinationHostname, "(?i)api\.skycloudcenter\.com"), "C2 Domain: api.skycloudcenter.com",
    match(DestinationHostname, "(?i)api\.wiresguard\.com"), "C2 Domain: api.wiresguard.com (WireGuard typosquat)",
    match(DestinationHostname, "(?i)cdncheck\.it\.com"), "C2 Domain: cdncheck.it.com",
    match(DestinationHostname, "(?i)^temp\.sh$"), "C2 Domain: temp.sh (file staging)",
    match(DestinationHostname, "(?i)self-dns\.it\.com"), "C2 Domain: self-dns.it.com",
    match(DestinationHostname, "(?i)safe-dns\.it\.com"), "C2 Domain: safe-dns.it.com",
    match(DestinationHostname, "(?i)cloudtrafficservice\.com"), "C2 Domain: cloudtrafficservice.com",
    1==1, "Network IOC Match"
)
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| eval mitre="T1071 - Application Layer Protocol, T1573 - Encrypted Channel, T1105 - Ingress Tool Transfer"
| eval port_note=if(
    (DestinationIp=="59.110.7.32" AND DestinationPort=="8880") OR (DestinationIp=="124.222.137.114" AND DestinationPort=="9999"),
    "CONFIRMED: Known C2 IP:Port combination",
    "C2 IP match (check port for additional context)"
)
| table _time, host, User, severity, ioc_match, port_note, mitre, Image, DestinationIp, DestinationPort, DestinationHostname, SourceIp, SourcePort,
| sort - _time
```

Search 26 — FIREWALL LOGS: C2 IP Detection (OPNsense / ZenArmor / Generic FW)

Detects connections to C2 IPs in firewall and NGFW logs. Covers: OPNsense filterlog, ZenArmor connection logs, and generic firewall sourcetypes.

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol, T1573 - Encrypted Channel | **Data Sources:** OPNsense, ZenArmor, Palo Alto, Cisco ASA, FortiGate

SPL — Search 26

```
index=* sourcetype IN ("opnsense:filterlog", "zenarmor:conn", "zenarmor:alert", "pan:traffic", "cisco:asa", "fortinet:fortigate", "firewall")
    (
        "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
        OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
        OR "103.159.133.178"
    )
| eval dest_ip=coalesce(dest_ip, dst_ip, DestinationIP, dest, dst, DstIP)
| eval src_ip=coalesce(src_ip, src, SrcIP, SourceIP)
| eval dest_port=coalesce(dest_port, dst_port, DestinationPort, dpt, DstPort)
| eval fw_action=coalesce(action, Action, fw_action, "unknown")
| eval ioc_match=case(
    dest_ip=="45.76.155.202", "C2: 45.76.155.202 (Vultr - Lotus Blossom)",
    dest_ip=="45.77.31.210", "C2: 45.77.31.210 (Vultr - Lotus Blossom)",
    dest_ip=="59.110.7.32", "C2: 59.110.7.32 (Alibaba - Chrysalis, port 8880)",
    dest_ip=="61.4.102.97", "C2: 61.4.102.97 (Lotus Blossom)",
    dest_ip=="95.179.213.0", "C2: 95.179.213.0 (Vultr - Lotus Blossom)",
    dest_ip=="124.222.137.114", "C2: 124.222.137.114 (Tencent - Chrysalis, port 9999)",
    dest_ip=="45.32.144.255", "C2: 45.32.144.255 (Vultr - Lotus Blossom)",
    dest_ip=="160.250.93.48", "C2: 160.250.93.48 (Lotus Blossom)",
    dest_ip=="103.159.133.178", "C2: 103.159.133.178 (Lotus Blossom)",
    src_ip=="45.76.155.202", "C2 INBOUND: 45.76.155.202 (Vultr - Lotus Blossom)",
    src_ip=="45.77.31.210", "C2 INBOUND: 45.77.31.210 (Vultr - Lotus Blossom)",
    src_ip=="59.110.7.32", "C2 INBOUND: 59.110.7.32 (Alibaba - Chrysalis)",
    src_ip=="61.4.102.97", "C2 INBOUND: 61.4.102.97 (Lotus Blossom)",
    src_ip=="95.179.213.0", "C2 INBOUND: 95.179.213.0 (Vultr - Lotus Blossom)",
    src_ip=="124.222.137.114", "C2 INBOUND: 124.222.137.114 (Tencent - Chrysalis)",
    src_ip=="45.32.144.255", "C2 INBOUND: 45.32.144.255 (Vultr - Lotus Blossom)",
    src_ip=="160.250.93.48", "C2 INBOUND: 160.250.93.48 (Lotus Blossom)",
    src_ip=="103.159.133.178", "C2 INBOUND: 103.159.133.178 (Lotus Blossom)",
    1==1, "Network IOC Match (check raw event)"
)
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| eval direction=if(match(ioc_match, "INBOUND"), "INBOUND (scanning/callback)", "OUTBOUND (C2 communication)")
| table _time, host, severity, ioc_match, direction, src_ip, dest_ip, dest_port, fw_action, sourcetype
| sort - _time
```

Search 27 — CIM NETWORK TRAFFIC DATA MODEL: C2 IP Detection (CIM-compliant)

Uses Splunk CIM Network_Traffic data model for normalized detection across all CIM-compliant network data sources (firewalls, IDS/IPS, proxies, Sysmon, etc.).

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol, T1573 - Encrypted Channel | **Data Sources:** CIM Network_Traffic.All_Traffic Data Model

SPL — Search 27

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(All_Traffic.action) as actions, values(All_Tr
    FROM datamodel=Network_Traffic.All_Traffic
    WHERE (
        All_Traffic.dest IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "16
        OR All_Traffic.src IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "
    )
    BY All_Traffic.dest, All_Traffic.dest_port, All_Traffic.transport, All_Traffic.src
| rename "All_Traffic.*" as *
| eval ioc_match=case(
    dest=="45.76.155.202", "C2: 45.76.155.202 (Vultr)",
    dest=="45.77.31.210", "C2: 45.77.31.210 (Vultr)",
    dest=="59.110.7.32", "C2: 59.110.7.32 (Alibaba, port 8880)",
    dest=="61.4.102.97", "C2: 61.4.102.97",
    dest=="95.179.213.0", "C2: 95.179.213.0 (Vultr)",
    dest=="124.222.137.114", "C2: 124.222.137.114 (Tencent, port 9999)",
    dest=="45.32.144.255", "C2: 45.32.144.255 (Vultr)",
    dest=="160.250.93.48", "C2: 160.250.93.48",
    dest=="103.159.133.178", "C2: 103.159.133.178",
    src IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.93.48", "10
    1==1, "Network IOC Match"
)
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| eval mitre="T1071 - Application Layer Protocol, T1573 - Encrypted Channel"
| table first_seen, last_seen, severity, ioc_match, threat_group, mitre, src, sources, dest, dest_port, transport, actions, apps, count
| sort - last_seen
```

Search 28a — DNS RESOLUTION: Sysmon DNS Query Logging (EventCode 22)

Detects DNS queries for known Lotus Blossom C2 domains via Sysmon EventCode 22 (DNSEvent).

Severity: CRITICAL | **MITRE:** T1071.004 - DNS, T1568 - Dynamic Resolution, T1583.001 - Domains | **Data Sources:** Sysmon (EventCode 22 - DNSEvent)

SPL — Search 28a

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=22
    (
        "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval queried_domain=QueryName
| eval ioc_match=case(
    match(queried_domain, "(?i)api\.skycloudcenter\.com"), "C2 DNS: api.skycloudcenter.com",
    match(queried_domain, "(?i)skycloudcenter\.com"), "C2 DNS: skycloudcenter.com (parent domain)",
    match(queried_domain, "(?i)api\.wiresguard\.com"), "C2 DNS: api.wiresguard.com (WireGuard typosquat)",
    match(queried_domain, "(?i)wiresguard\.com"), "C2 DNS: wiresguard.com (parent domain, WireGuard typosquat)",
    match(queried_domain, "(?i)cdncheck\.it\.com"), "C2 DNS: cdncheck.it.com",
    match(queried_domain, "(?i)^temp\.sh$"), "C2 DNS: temp.sh (file staging service)",
    match(queried_domain, "(?i)self-dns\.it\.com"), "C2 DNS: self-dns.it.com",
    match(queried_domain, "(?i)safe-dns\.it\.com"), "C2 DNS: safe-dns.it.com",
    match(queried_domain, "(?i)cloudtrafficservice\.com"), "C2 DNS: cloudtrafficservice.com",
    1==1, "C2 Domain DNS Query"
)
| eval severity="CRITICAL"
| eval mitre="T1071.004 - DNS, T1568 - Dynamic Resolution, T1583.001 - Domains"
| eval resolved_ip=coalesce(QueryResults, "N/A")
| table _time, host, User, severity, ioc_match, mitre, queried_domain, resolved_ip, Image, ProcessId
| sort - _time
```

Search 28b — DNS RESOLUTION: CIM Network Resolution Data Model

Uses CIM Network_Resolution.DNS data model for normalized DNS detection.

Severity: CRITICAL | **MITRE:** T1071.004 - DNS, T1568 - Dynamic Resolution | **Data Sources:** CIM Network_Resolution.DNS Data Model

SPL — Search 28b

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(DNS.src) as sources, values(DNS.answer) as an
    FROM datamodel=Network_Resolution.DNS
    WHERE (
        DNS.query IN ("api.skycloudcenter.com", "api.wiresguard.com", "cdncheck.it.com", "temp.sh", "self-dns.it.com", "safe-dns.it.com", "cloudtra
        OR DNS.query="*.skycloudcenter.com"
        OR DNS.query="*.wiresguard.com"
        OR DNS.query="*.cloudtrafficservice.com"
        OR DNS.query="*-dns.it.com"
        OR DNS.query="*check.it.com"
    )
    BY DNS.query, DNS.message_type
| rename "DNS.*" as *
| eval ioc_match=case(
    match(query, "(?i)skycloudcenter\.com"), "C2 Domain: skycloudcenter.com",
    match(query, "(?i)wiresguard\.com"), "C2 Domain: wiresguard.com (WireGuard typosquat)",
    match(query, "(?i)cdncheck\.it\.com"), "C2 Domain: cdncheck.it.com",
    match(query, "(?i)^temp\.sh$"), "C2 Domain: temp.sh",
    match(query, "(?i)self-dns\.it\.com"), "C2 Domain: self-dns.it.com",
    match(query, "(?i)safe-dns\.it\.com"), "C2 Domain: safe-dns.it.com",
    match(query, "(?i)cloudtrafficservice\.com"), "C2 Domain: cloudtrafficservice.com",
    1==1, "C2 Domain DNS Match"
)
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| table first_seen, last_seen, severity, ioc_match, threat_group, query, message_type, sources, answers, count
| sort - last_seen
```

Search 28c — DNS RESOLUTION: Windows DNS Client Event Log

Detects C2 domain DNS queries in Windows DNS Client operational log.

Severity: CRITICAL | **MITRE:** T1071.004 - DNS | **Data Sources:** Windows DNS Client Operational Log

SPL — Search 28c

```
index=* sourcetype IN ("WinEventLog:Microsoft-Windows-DNS-Client/Operational", "XmlWinEventLog:Microsoft-Windows-DNS-Client/Operational")
    (
        "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval ioc_match="CRITICAL: DNS Query for Lotus Blossom C2 Domain"
| eval threat_group="Lotus Blossom (Chinese APT)"
| table _time, host, EventCode, ioc_match, threat_group, QueryName, QueryType, QueryResults
| sort - _time
```

Search 28d — DNS RESOLUTION: DNS Server Logs

Detects C2 domain queries in DNS server logs (Windows DNS, BIND, Infoblox).

Severity: CRITICAL | **MITRE:** T1071.004 - DNS | **Data Sources:** DNS Server Logs (Windows DNS, BIND, Infoblox)

SPL — Search 28d

```
index=* sourcetype IN ("WinEventLog:DNS Server", "XmlWinEventLog:DNS Server", "dns", "named", "bind", "infoblox:dns")
    (
        "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval queried=coalesce(query, QNAME, QueryName, "N/A")
| eval client=coalesce(src_ip, ClientIP, src, "N/A")
| eval ioc_match="CRITICAL: DNS Resolution of Lotus Blossom C2 Domain"
| table _time, host, client, ioc_match, queried, sourcetype
| sort - _time
```

Search 29 — PROXY / WEB LOGS: C2 Domain and IP Detection

Detects C2 communication in web proxy logs (Squid, Blue Coat, Zscaler, etc.). Also covers ZenArmor HTTP logs with full URL and domain data.

Severity: CRITICAL | **MITRE:** T1071.001 - Web Protocols, T1090 - Proxy | **Data Sources:** ZenArmor HTTP, Squid, Blue Coat, Zscaler, Generic Proxy

SPL — Search 29

```
index=* sourcetype IN ("zenarmor:http", "squid:access", "bluecoat:proxysg", "zscaler:web", "proxy", "websense", "mcafee:wg")
    (
        "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
        OR "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
        OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
        OR "103.159.133.178"
    )
| eval dest_domain=coalesce(url_domain, dest_host, http_host, host_header, site, "N/A")
| eval dest_ip_field=coalesce(dest_ip, dst_ip, dest, "N/A")
| eval request_url=coalesce(url, uri, request_uri, cs_uri_stem, "N/A")
| eval user_agent=coalesce(http_user_agent, cs_User_Agent, useragent, "N/A")
| eval proxy_action=coalesce(action, sc_filter_result, status, "N/A")
| eval ioc_match=case(
    match(dest_domain, "(?i)skycloudcenter\.com"), "C2 Proxy: skycloudcenter.com",
    match(dest_domain, "(?i)wiresguard\.com"), "C2 Proxy: wiresguard.com (typosquat)",
    match(dest_domain, "(?i)cdncheck\.it\.com"), "C2 Proxy: cdncheck.it.com",
    match(dest_domain, "(?i)^temp\.sh$"), "C2 Proxy: temp.sh (file staging)",
    match(dest_domain, "(?i)self-dns\.it\.com"), "C2 Proxy: self-dns.it.com",
    match(dest_domain, "(?i)safe-dns\.it\.com"), "C2 Proxy: safe-dns.it.com",
    match(dest_domain, "(?i)cloudtrafficservice\.com"), "C2 Proxy: cloudtrafficservice.com",
    dest_ip_field IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.9
    1==1, "Network IOC Proxy Match"
)
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| table _time, host, severity, ioc_match, threat_group, dest_domain, dest_ip_field, request_url, user_agent, proxy_action, sourcetype
| sort - _time
```

Search 30 — CIM WEB DATA MODEL: C2 Domain/IP Detection (CIM-compliant)

Uses the Splunk CIM Web data model for normalized proxy/web detection.

Severity: CRITICAL | **MITRE:** T1071.001 - Web Protocols | **Data Sources:** CIM Web.Web Data Model

SPL — Search 30

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(Web.src) as sources, values(Web.http_method)
    FROM datamodel=Web.Web
    WHERE (
        Web.url IN ("*skycloudcenter.com*", "*wiresguard.com*", "*cdncheck.it.com*", "*temp.sh*", "*self-dns.it.com*", "*safe-dns.it.com*", "*cloud
        OR Web.dest IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250
    )
    BY Web.dest, Web.url, Web.http_user_agent
| rename "Web.*" as *
| eval severity="CRITICAL"
| eval threat_group="Lotus Blossom (Chinese APT)"
| eval ioc_match=case(
    match(url, "(?i)skycloudcenter\.com"), "C2: skycloudcenter.com",
    match(url, "(?i)wiresguard\.com"), "C2: wiresguard.com",
    match(url, "(?i)cdncheck\.it\.com"), "C2: cdncheck.it.com",
    match(url, "(?i)temp\.sh"), "C2: temp.sh",
    match(url, "(?i)self-dns\.it\.com"), "C2: self-dns.it.com",
    match(url, "(?i)safe-dns\.it\.com"), "C2: safe-dns.it.com",
    match(url, "(?i)cloudtrafficservice\.com"), "C2: cloudtrafficservice.com",
    dest IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.93.48", "1
    1==1, "Network IOC Match"
)
| table first_seen, last_seen, severity, ioc_match, threat_group, sources, dest, url, http_user_agent, methods, statuses, count
| sort - last_seen
```

Search 31 — ZENARMOR NGFW: Comprehensive C2 Detection across all ZenArmor sourcetypes

Leverages all 5 ZenArmor sourcetypes for deep network inspection: zenarmor:conn (Connection-level), zenarmor:dns (DNS resolution), zenarmor:http (HTTP traffic), zenarmor:tls (TLS SNI/certificate), zenarmor:alert (IDS/IPS alerts).

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol, T1573 - Encrypted Channel | **Data Sources:** ZenArmor (conn, dns, http, tls, alert)

SPL — Search 31

```
index=* sourcetype IN ("zenarmor:conn", "zenarmor:dns", "zenarmor:http", "zenarmor:tls", "zenarmor:alert")
    (
        "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
        OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
        OR "103.159.133.178"
        OR "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval dest_ip_field=coalesce(dest_ip, dst_ip, DestinationIP, "N/A")
| eval dest_domain=coalesce(url_domain, query, sni, http_host, dest_host, "N/A")
| eval ioc_type=case(
    sourcetype=="zenarmor:conn", "Connection",
    sourcetype=="zenarmor:dns", "DNS Resolution",
    sourcetype=="zenarmor:http", "HTTP Request",
    sourcetype=="zenarmor:tls", "TLS Handshake",
    sourcetype=="zenarmor:alert", "IDS/IPS Alert",
    1==1, sourcetype
)
| eval ioc_match=case(
    dest_ip_field IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.9
    match(dest_domain, "(?i)(skycloudcenter|wiresguard|cdncheck\.it|self-dns\.it|safe-dns\.it|cloudtrafficservice)\.com"), "C2 Domain: ".dest_domai
    match(dest_domain, "(?i)^temp\.sh$"), "C2 Domain: temp.sh",
    1==1, "Network IOC Match"
)
| eval severity="CRITICAL"
| table _time, host, severity, ioc_type, ioc_match, sourcetype, src_ip, dest_ip_field, dest_domain, dest_port
| sort - _time
```

Search 32a — ZENARMOR TLS/CERTIFICATE: C2 Domain Detection via TLS SNI

Detects C2 domains in TLS Server Name Indication (SNI) fields and certificate subjects. Catches encrypted C2 traffic even when HTTP content is not visible.

Severity: CRITICAL | **MITRE:** T1573.002 - Asymmetric Cryptography, T1071 - Application Layer Protocol | **Data Sources:** ZenArmor TLS (zenarmor:tls)

SPL — Search 32a

```
index=* sourcetype="zenarmor:tls"
    (
        "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval sni_domain=coalesce(sni, ssl_subject_common_name, ssl_issuer_common_name, "N/A")
| eval ioc_match=case(
    match(sni_domain, "(?i)skycloudcenter\.com"), "TLS SNI: skycloudcenter.com (C2)",
    match(sni_domain, "(?i)wiresguard\.com"), "TLS SNI: wiresguard.com (C2 typosquat)",
    match(sni_domain, "(?i)cdncheck\.it\.com"), "TLS SNI: cdncheck.it.com (C2)",
    match(sni_domain, "(?i)self-dns\.it\.com"), "TLS SNI: self-dns.it.com (C2)",
    match(sni_domain, "(?i)safe-dns\.it\.com"), "TLS SNI: safe-dns.it.com (C2)",
    match(sni_domain, "(?i)cloudtrafficservice\.com"), "TLS SNI: cloudtrafficservice.com (C2)",
    1==1, "TLS C2 Domain Match"
)
| eval severity="CRITICAL"
| table _time, host, severity, ioc_match, src_ip, dest_ip, dest_port, sni_domain, ssl_version, ssl_cipher
| sort - _time
```

Search 32b — CIM CERTIFICATES DATA MODEL: C2 Domain Detection

Uses CIM Certificates.All_Certificates data model for normalized TLS certificate detection of C2 domains.

Severity: CRITICAL | **MITRE:** T1573.002 - Asymmetric Cryptography | **Data Sources:** CIM Certificates.All_Certificates Data Model

SPL — Search 32b

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(All_Certificates.src) as sources
    FROM datamodel=Certificates.All_Certificates
    WHERE (
        All_Certificates.ssl_subject_common_name IN ("*skycloudcenter.com*", "*wiresguard.com*", "*cdncheck.it.com*", "*self-dns.it.com*", "*safe-d
        OR All_Certificates.ssl_issuer_common_name IN ("*skycloudcenter.com*", "*wiresguard.com*", "*cdncheck.it.com*", "*self-dns.it.com*", "*safe
    )
    BY All_Certificates.ssl_subject_common_name, All_Certificates.ssl_issuer_common_name, All_Certificates.dest
| rename "All_Certificates.*" as *
| eval severity="CRITICAL"
| eval ioc_match="TLS Certificate Match for Lotus Blossom C2 Domain"
| table first_seen, last_seen, severity, ioc_match, ssl_subject_common_name, ssl_issuer_common_name, dest, sources, count
| sort - last_seen
```

Search 33a — SUSPICIOUS USER-AGENT: High-Confidence (UA + C2 destination)

High-confidence detection combining known Chrysalis implant User-Agent strings with C2 destination IPs/domains. Chrome/80.0.4044.92 (Feb 2020), Chrome/92.0.4472.114 (Aug 2021), Chrome/134.0.0.0 (spoofed).

Severity: CRITICAL | **MITRE:** T1071.001 - Web Protocols, T1036.005 - Match Legitimate Name or Location | **Data Sources:** ZenArmor HTTP, Squid, Blue Coat, Zscaler, Sysmon

SPL — Search 33a

```
index=* sourcetype IN ("zenarmor:http", "squid:access", "bluecoat:proxysg", "zscaler:web", "proxy", "XmlWinEventLog:Microsoft-Windows-Sysmon/Operat
    (
        ("Chrome/80.0.4044.92" OR "Chrome/92.0.4472.114" OR "Chrome/134.0.0.0")
        AND (
            "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
            OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
            OR "103.159.133.178"
            OR "skycloudcenter" OR "wiresguard" OR "cdncheck.it" OR "temp.sh"
            OR "self-dns.it" OR "safe-dns.it" OR "cloudtrafficservice"
        )
    )
| eval user_agent=coalesce(http_user_agent, cs_User_Agent, useragent, "N/A")
| eval dest_combined=coalesce(dest_ip, DestinationIp, dest, url_domain, DestinationHostname, "N/A")
| eval ioc_match="CRITICAL: Chrysalis Implant User-Agent + C2 Destination"
| eval ua_type=case(
    match(user_agent, "Chrome/80\.0\.4044\.92"), "Chrome/80 (Feb 2020 - very outdated)",
    match(user_agent, "Chrome/92\.0\.4472\.114"), "Chrome/92 (Aug 2021 - outdated, macOS spoof)",
    match(user_agent, "Chrome/134\.0\.0\.0"), "Chrome/134 (potentially spoofed version)",
    1==1, "Unknown suspicious UA"
)
| eval severity="CRITICAL"
| eval mitre="T1071.001 - Web Protocols, T1036.005 - Match Legitimate Name or Location"
| table _time, host, severity, ioc_match, ua_type, user_agent, dest_combined, sourcetype
| sort - _time
```

Search 33b — SUSPICIOUS USER-AGENT: Threat Hunting (broader, more FPs)

Medium-confidence search for periodic threat hunting. Detects outdated Chrome UAs that may indicate Chrysalis implant. Filter out known legitimate outdated browsers.

Severity: MEDIUM | **MITRE:** T1071.001 - Web Protocols | **Data Sources:** ZenArmor HTTP, Squid, Blue Coat, Zscaler, Proxy

SPL — Search 33b

```
index=* sourcetype IN ("zenarmor:http", "squid:access", "bluecoat:proxysg", "zscaler:web", "proxy")
    ("Chrome/80.0.4044.92" OR "Chrome/92.0.4472.114")
| eval user_agent=coalesce(http_user_agent, cs_User_Agent, useragent, "N/A")
| eval src_host=coalesce(src_ip, src, c_ip, "N/A")
| eval dest_combined=coalesce(dest_ip, url_domain, dest_host, dest, "N/A")
| eval dest_url=coalesce(url, uri, cs_uri_stem, "N/A")
| where NOT match(dest_combined, "(?i)(microsoft\.com|windows\.com|windowsupdate\.com|google\.com|googleapis\.com|gstatic\.com|chrome\.com)")
| stats count as requests, dc(dest_combined) as unique_destinations, earliest(_time) as first_seen, latest(_time) as last_seen, values(dest_combine
    BY src_host, user_agent
| where requests > 1
| eval first_seen=strftime(first_seen, "%Y-%m-%d %H:%M:%S")
| eval last_seen=strftime(last_seen, "%Y-%m-%d %H:%M:%S")
| eval investigation_note="THREAT HUNT: Verify if this host legitimately runs an outdated Chrome browser. If not, this is a strong Chrysalis implan
| sort - requests
```

PART C — Supply Chain / GUP.exe Detection

Search 34a — GUP.EXE: Non-legitimate network connections (Sysmon)

GUP.exe (Notepad++ updater) making connections to non-legitimate destinations. Lotus Blossom abused the Notepad++ supply chain, trojanizing GUP.exe or using it for DLL side-loading.

Severity: CRITICAL / HIGH | **MITRE:** T1195.002 - Supply Chain Compromise, T1071.001 - Web Protocols | **Data Sources:** Sysmon (EventCode 3 - NetworkConnect)

SPL — Search 34a

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=3
    ("gup.exe" OR "GUP.exe")
    Initiated="true"
| eval proc=Image
| where match(proc, "(?i)\\\\(gup|GUP)\.exe$")
| where NOT match(DestinationHostname, "(?i)^(notepad-plus-plus\.org|github\.com|release-assets\.githubusercontent\.com|objects\.githubusercontent\
    AND NOT DestinationIp IN ("127.0.0.1", "0.0.0.0", "::1")
    AND NOT match(DestinationIp, "^(10\.|172\.(1[6-9]|2[0-9]|3[01])\.|192\.168\.)")
| eval ioc_match=case(
    DestinationIp IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.9
        "CRITICAL: GUP.exe connecting to KNOWN Lotus Blossom C2 IP: ".DestinationIp,
    match(DestinationHostname, "(?i)(skycloudcenter|wiresguard|cdncheck\.it|self-dns\.it|safe-dns\.it|cloudtrafficservice|temp\.sh)"),
        "CRITICAL: GUP.exe connecting to KNOWN Lotus Blossom C2 domain: ".DestinationHostname,
    1==1,
        "HIGH: GUP.exe (Notepad++ updater) connecting to non-legitimate destination: ".coalesce(DestinationHostname, DestinationIp)
)
| eval severity=if(match(ioc_match, "CRITICAL"), "CRITICAL", "HIGH")
| eval mitre="T1195.002 - Supply Chain Compromise, T1071.001 - Web Protocols"
| eval investigation="Verify Notepad++ installation integrity. Check DLLs in Notepad++ directory. Compare GUP.exe hash against known-good."
| table _time, host, User, severity, ioc_match, mitre, proc, DestinationIp, DestinationPort, DestinationHostname, CommandLine, ParentImage, Hashes,
| sort - _time
```

Search 34b — GUP.EXE: Spawning suspicious child processes

Detects GUP.exe spawning unexpected child processes, which is abnormal behavior and may indicate a trojanized Notepad++ updater.

Severity: CRITICAL / HIGH / MEDIUM | **MITRE:** T1195.002 - Supply Chain Compromise, T1059 - Command and Scripting Interpreter | **Data Sources:** Sysmon (EventCode 1 - ProcessCreate)

SPL — Search 34b

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=1
    ParentImage="*gup.exe"
| eval parent=ParentImage
| eval child=Image
| eval cmdline=CommandLine
| where match(parent, "(?i)\\\\(gup|GUP)\.exe$")
    AND NOT match(child, "(?i)\\\\conhost\.exe$")
| eval ioc_match=case(
    match(child, "(?i)\\\\(USOShared|ProShow|Adobe\\\\Scripts|Bluetooth)\\\\"), "CRITICAL: GUP.exe spawned Lotusblossom implant process",
    match(child, "(?i)\\\\(cmd|powershell|pwsh|wscript|cscript|mshta|certutil|bitsadmin|rundll32|regsvr32)\.exe$"), "HIGH: GUP.exe spawned suspicio
    match(child, "(?i)\\\\(AppData|Temp|ProgramData)\\\\"), "HIGH: GUP.exe spawned process from suspicious path",
    1==1, "MEDIUM: GUP.exe spawned unexpected child process"
)
| eval severity=case(match(ioc_match, "CRITICAL"), "CRITICAL", match(ioc_match, "HIGH"), "HIGH", 1==1, "MEDIUM")
| eval mitre="T1195.002 - Supply Chain Compromise, T1059 - Command and Scripting Interpreter"
| table _time, host, User, severity, ioc_match, mitre, parent, child, cmdline, ParentCommandLine, Hashes, ProcessId, ParentProcessId
| sort - _time
```

Search 34c — GUP.EXE: Loading unexpected DLLs (DLL side-loading)

Detects GUP.exe loading DLLs from non-standard locations, indicating potential DLL side-loading attack vector used by Lotus Blossom.

Severity: CRITICAL / HIGH / MEDIUM | **MITRE:** T1574.002 - DLL Side-Loading, T1195.002 - Supply Chain Compromise | **Data Sources:** Sysmon (EventCode 7 - ImageLoad)

SPL — Search 34c

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=7
    ("gup.exe" OR "GUP.exe")
| eval loading_process=Image
| eval loaded_dll=ImageLoaded
| where match(loading_process, "(?i)\\\\(gup|GUP)\.exe$")
    AND NOT match(loaded_dll, "(?i)\\\\(Windows\\\\|System32\\\\|SysWOW64\\\\|WinSxS\\\\|notepad\+\+\\\\|Program Files)")
    AND NOT match(loaded_dll, "(?i)\\\\(ntdll|kernel32|kernelbase|msvcrt|ucrtbase|advapi32|sechost|rpcrt4|combase|oleaut32|ole32|user32|gdi32|shell
| eval ioc_match=case(
    match(loaded_dll, "(?i)\\\\libtcc\.dll$"), "CRITICAL: GUP.exe loaded libtcc.dll (Lotusblossom indicator)",
    match(loaded_dll, "(?i)\\\\(USOShared|AppData\\\\Roaming\\\\(ProShow|Adobe\\\\Scripts|Bluetooth))\\\\"), "CRITICAL: GUP.exe loaded DLL from Lot
    coalesce(Signed, "false")=="false", "HIGH: GUP.exe loaded unsigned DLL: ".loaded_dll,
    coalesce(SignatureStatus, "")!="Valid", "HIGH: GUP.exe loaded DLL with invalid signature: ".loaded_dll,
    1==1, "MEDIUM: GUP.exe loaded unexpected DLL: ".loaded_dll
)
| eval severity=case(match(ioc_match, "CRITICAL"), "CRITICAL", match(ioc_match, "HIGH"), "HIGH", 1==1, "MEDIUM")
| eval mitre="T1574.002 - DLL Side-Loading, T1195.002 - Supply Chain Compromise"
| table _time, host, User, severity, ioc_match, mitre, loading_process, loaded_dll, Signed, SignatureStatus, Hashes, ProcessId
| sort - _time
```

Search 34d — GUP.EXE: Non-standard command line arguments

Detects GUP.exe launched with unusual command line arguments, which may indicate it has been weaponized to contact C2 infrastructure.

Severity: CRITICAL / HIGH / MEDIUM | **MITRE:** T1195.002 - Supply Chain Compromise, T1059 - Command and Scripting Interpreter | **Data Sources:** Sysmon (EventCode 1 - ProcessCreate)

SPL — Search 34d

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational")
    EventCode=1
    ("gup.exe" OR "GUP.exe")
| eval proc=Image
| eval cmdline=CommandLine
| where match(proc, "(?i)\\\\(gup|GUP)\.exe$")
| where NOT match(cmdline, "(?i)(notepad-plus-plus\.org|github\.com/notepad-plus-plus)")
    AND NOT match(cmdline, "(?i)\\\\notepad\+\+\\\\updater\\\\")
    AND NOT match(cmdline, "(?i)^\"[^\"]+\\\\gup\.exe\"$")
    AND len(cmdline) > 0
| eval ioc_match=case(
    match(cmdline, "(?i)(skycloudcenter|wiresguard|cdncheck|self-dns|safe-dns|cloudtrafficservice|temp\.sh)"), "CRITICAL: GUP.exe launched with C2
    match(cmdline, "(?i)(45\.76\.155\.202|45\.77\.31\.210|59\.110\.7\.32|61\.4\.102\.97|95\.179\.213\.0|124\.222\.137\.114|45\.32\.144\.255|160\.25
    match(cmdline, "(?i)(http|https)://") AND NOT match(cmdline, "(?i)notepad-plus-plus\.org"), "HIGH: GUP.exe launched with non-standard URL",
    1==1, "MEDIUM: GUP.exe launched with unusual command line"
)
| eval severity=case(match(ioc_match, "CRITICAL"), "CRITICAL", match(ioc_match, "HIGH"), "HIGH", 1==1, "MEDIUM")
| eval mitre="T1195.002 - Supply Chain Compromise, T1059 - Command and Scripting Interpreter"
| table _time, host, User, severity, ioc_match, mitre, proc, cmdline, ParentImage, ParentCommandLine, Hashes, ProcessId
| sort - _time
```

PART D — High-Confidence Alerts & Dashboards

Search 35 — NON-STANDARD PORT C2: Specific IP:Port combinations

Two C2 servers use non-standard ports: 59.110.7.32:8880 (Alibaba Cloud) and 124.222.137.114:9999 (Tencent Cloud). Very high confidence - exact IP:Port match from campaign intelligence.

Severity: CRITICAL | **MITRE:** T1571 - Non-Standard Port, T1071 - Application Layer Protocol, T1573 - Encrypted Channel | **Data Sources:** Sysmon, OPNsense, ZenArmor

SPL — Search 35

```
index=* sourcetype IN ("XmlWinEventLog:Microsoft-Windows-Sysmon/Operational", "xmlwineventlog:microsoft-windows-sysmon/operational", "opnsense:filt
    (
        ("59.110.7.32" AND "8880")
        OR ("124.222.137.114" AND "9999")
    )
| eval dest_ip_field=coalesce(DestinationIp, dest_ip, dst_ip, "N/A")
| eval dest_port_field=coalesce(DestinationPort, dest_port, dst_port, "N/A")
| where (dest_ip_field=="59.110.7.32" AND dest_port_field=="8880")
    OR (dest_ip_field=="124.222.137.114" AND dest_port_field=="9999")
| eval ioc_match=case(
    dest_ip_field=="59.110.7.32" AND dest_port_field=="8880", "CRITICAL: Confirmed C2 - 59.110.7.32:8880 (Alibaba Cloud - Chrysalis C2)",
    dest_ip_field=="124.222.137.114" AND dest_port_field=="9999", "CRITICAL: Confirmed C2 - 124.222.137.114:9999 (Tencent Cloud - Chrysalis C2)",
    1==1, "C2 IP:Port Match"
)
| eval severity="CRITICAL"
| eval confidence="VERY HIGH - Exact IP:Port match from campaign intelligence"
| eval response="IMMEDIATE ISOLATION: Confirmed Lotus Blossom C2 communication. Capture memory, preserve network logs, initiate full IR."
| eval mitre="T1571 - Non-Standard Port, T1071 - Application Layer Protocol, T1573 - Encrypted Channel"
| table _time, host, severity, confidence, ioc_match, mitre, response, Image, src_ip, dest_ip_field, dest_port_field, sourcetype, ProcessId
| sort - _time
```

Search 36a — INTRUSION DETECTION: ZenArmor/Suricata IDS alerts for C2 traffic

Leverages IDS/IPS alerts from ZenArmor (Suricata-based) for detection of C2 traffic patterns.

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol | **Data Sources:** ZenArmor IDS/IPS Alerts (zenarmor:alert)

SPL — Search 36a

```
index=* sourcetype="zenarmor:alert"
    (
        "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
        OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
        OR "103.159.133.178"
        OR "skycloudcenter" OR "wiresguard" OR "cdncheck" OR "self-dns"
        OR "safe-dns" OR "cloudtrafficservice"
    )
| eval severity="CRITICAL"
| eval ioc_match="IDS Alert matching Lotus Blossom C2 infrastructure"
| table _time, host, severity, ioc_match, alert_signature, alert_category, alert_severity, src_ip, dest_ip, dest_port, proto
| sort - _time
```

Search 36b — INTRUSION DETECTION: CIM Intrusion Detection Data Model

Uses CIM Intrusion_Detection.IDS_Attacks data model for normalized IDS detection of C2 traffic.

Severity: CRITICAL | **MITRE:** T1071 - Application Layer Protocol | **Data Sources:** CIM Intrusion_Detection.IDS_Attacks Data Model

SPL — Search 36b

```
| tstats summariesonly=false count, earliest(_time) as first_seen, latest(_time) as last_seen, values(IDS_Attacks.signature) as signatures, values(
    FROM datamodel=Intrusion_Detection.IDS_Attacks
    WHERE (
        IDS_Attacks.dest IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "16
        OR IDS_Attacks.src IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "
    )
    BY IDS_Attacks.dest, IDS_Attacks.src, IDS_Attacks.category
| rename "IDS_Attacks.*" as *
| eval severity="CRITICAL"
| eval ioc_match="IDS Detection matching Lotus Blossom C2"
| table first_seen, last_seen, severity, ioc_match, src, dest, category, signatures, severities, count
| sort - last_seen
```

Search 37 — NETWORK IOC LOOKUP TABLE: CSV-based approach for production

CSV lookup for all network IOCs. Import into Splunk for efficient matching. For production: outputlookup then use with CIM data models via automatic lookups.

Severity: CRITICAL / HIGH | **MITRE:** Multiple | **Data Sources:** ZenArmor, OPNsense, Sysmon + lotusblossom_network_iocs.csv lookup

SPL — Search 37

```
` --- Run once to create the lookup: ---
| makeresults
| eval data="45.76.155.202,ip,Vultr VPS,Critical,Lotus Blossom C2:::45.77.31.210,ip,Vultr VPS,Critical,Lotus Blossom C2:::59.110.7.32,ip,Alibaba Cl
| makemv data delim=":::"
| mvexpand data
| rex field=data "(?<ioc_value>[^,]+),(?<ioc_type>[^,]+),(?<description>[^,]*),(?<severity>[^,]+),(?<threat_attribution>.+)"
| table ioc_value, ioc_type, description, severity, threat_attribution
| outputlookup lotusblossom_network_iocs.csv `

` --- Use the lookup for continuous network monitoring: ---
index=* sourcetype IN ("zenarmor:conn", "zenarmor:http", "zenarmor:dns", "zenarmor:tls", "opnsense:filterlog", "XmlWinEventLog:Microsoft-Windows-Sy
| eval dest_check=coalesce(DestinationIp, dest_ip, dst_ip, "N/A")
| eval domain_check=coalesce(DestinationHostname, url_domain, query, sni, http_host, dest_host, "N/A")
| lookup lotusblossom_network_iocs.csv ioc_value AS dest_check OUTPUT ioc_type, description, severity, threat_attribution
| eval ip_match=if(isnotnull(severity), "yes", "no")
| lookup lotusblossom_network_iocs.csv ioc_value AS domain_check OUTPUT ioc_type AS domain_ioc_type, description AS domain_description, severity AS
| eval domain_match=if(isnotnull(domain_severity), "yes", "no")
| where ip_match=="yes" OR domain_match=="yes"
| eval final_severity=coalesce(severity, domain_severity)
| eval final_ioc=coalesce(dest_check." (".description.")", domain_check." (".domain_description.")")
| eval final_threat=coalesce(threat_attribution, domain_threat)
| table _time, host, final_severity, final_ioc, final_threat, dest_check, domain_check, sourcetype
| sort - _time
```

Search 38 — COMBINED NETWORK + HOST ALERT: Full campaign detection (for Splunk Alert)

Single comprehensive alert combining ALL IOC types: Network IOCs (IPs, domains, ports), File IOCs (paths, filenames), Hash IOCs (SHA256), Behavioral IOCs (GUP.exe abuse, DLL side-loading). Recommended: Schedule every 5 minutes. Trigger on count > 0.

Severity: CRITICAL | **MITRE:** Multiple (Full Campaign Coverage) | **Data Sources:** Sysmon, OPNsense, ZenArmor, Windows Security

SPL — Search 38

```
index=* sourcetype IN (
    "XmlWinEventLog:Microsoft-Windows-Sysmon/Operational",
    "xmlwineventlog:microsoft-windows-sysmon/operational",
    "opnsense:filterlog",
    "zenarmor:conn", "zenarmor:dns", "zenarmor:http", "zenarmor:tls", "zenarmor:alert",
    "WinEventLog:Security", "wineventlog:security",
    "XmlWinEventLog:Security", "xmlwineventlog:security"
)
(
    "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
    OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
    OR "103.159.133.178"
    OR "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
    OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    OR "USOShared"
    OR ("Bluetooth" AND ("log.dll" OR "BluetoothService.exe"))
    OR ("ProShow" AND ("ProShow.exe" OR "load"))
    OR ("Adobe" AND "Scripts" AND ("alien.ini" OR "script.exe"))
    OR "libtcc.dll"
    OR "3bdc4c0637591533f1d4198a72a33426c01f69bd2e15ceee547866f65e26b7ad"
    OR "e2e3d78437cf9d48c2b2264e44bb36bc2235834fc45bbb50b5d6867f336711e3"
    OR "0a9b8df968df41920b6ff07785cbfebe8bda29e6b512c94a3b2a83d10014d2fd"
    OR "b4169a831292e245ebdffedd5820584d73b129411546e7d3eccf4663d5fc5be3"
    OR "29d0467ee452752286318f350ceb28a2b04ee4c6de550ba0edc34ae0fa7cbb03"
)
| eval target_path=coalesce(TargetFilename, Image, ImageLoaded, TargetObject, PipeName, "")
| eval hash_field=coalesce(Hashes, Hash, FileHash, "")
| eval sha256=lower(if(match(hash_field, "(?i)SHA256="), replace(hash_field, "(?i).*SHA256=([a-fA-F0-9]{64}).*", "\1"), ""))
| eval dest_ip_field=coalesce(DestinationIp, dest_ip, dst_ip, "")
| eval dest_domain=coalesce(DestinationHostname, url_domain, query, sni, dest_host, "")
| eval detection=case(
    sha256=="3bdc4c0637591533f1d4198a72a33426c01f69bd2e15ceee547866f65e26b7ad", "HASH: MAL_Chrysalis_DllLoader",
    sha256=="e2e3d78437cf9d48c2b2264e44bb36bc2235834fc45bbb50b5d6867f336711e3", "HASH: MAL_Chrysalis_Backdoor",
    sha256=="0a9b8df968df41920b6ff07785cbfebe8bda29e6b512c94a3b2a83d10014d2fd", "HASH: MAL_CobaltStrike_Loader_v1",
    sha256=="b4169a831292e245ebdffedd5820584d73b129411546e7d3eccf4663d5fc5be3", "HASH: MAL_CobaltStrike_Loader_v2",
    sha256=="29d0467ee452752286318f350ceb28a2b04ee4c6de550ba0edc34ae0fa7cbb03", "HASH: MAL_Warbird_Loader",
    dest_ip_field=="59.110.7.32" AND DestinationPort=="8880", "NETWORK: C2 59.110.7.32:8880 (Chrysalis)",
    dest_ip_field=="124.222.137.114" AND DestinationPort=="9999", "NETWORK: C2 124.222.137.114:9999 (Chrysalis)",
    dest_ip_field IN ("45.76.155.202", "45.77.31.210", "59.110.7.32", "61.4.102.97", "95.179.213.0", "124.222.137.114", "45.32.144.255", "160.250.9
    match(dest_domain, "(?i)(skycloudcenter|wiresguard|cdncheck\.it|self-dns\.it|safe-dns\.it|cloudtrafficservice)\.com"), "NETWORK: C2 Domain ".de
    match(dest_domain, "(?i)^temp\.sh$"), "NETWORK: C2 Domain temp.sh",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\log\.dll$"), "FILE: Chrysalis Loader (log.dll)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Bluetooth\\\\BluetoothService\.exe$"), "FILE: Chrysalis (BluetoothService.exe)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\ProShow\.exe$"), "FILE: Chrysalis (ProShow.exe)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\load$"), "FILE: Chrysalis Payload (load)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\script\.exe$"), "FILE: Chrysalis (script.exe)",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\alien\.ini$"), "FILE: Chrysalis Config (alien.ini)",
    match(target_path, "(?i)\\\\USOShared\\\\[a-zA-Z0-9]{1,15}\.(c|dll|exe)$"), "FILE: USOShared Suspicious",
    match(target_path, "(?i)\\\\libtcc\.dll$"), "FILE: libtcc.dll",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\ProShow\\\\[a-zA-Z0-9]{1}\.txt$"), "FILE: ProShow Staging TXT",
    match(target_path, "(?i)\\\\AppData\\\\Roaming\\\\Adobe\\\\Scripts\\\\[a-zA-Z0-9]{1}\.txt$"), "FILE: Adobe Scripts Staging TXT",
    1==1, null()
)
| where isnotnull(detection)
| eval severity=case(
    match(detection, "^HASH"), "CRITICAL",
    match(detection, "C2 59\.110.*8880|C2 124\.222.*9999"), "CRITICAL",
    match(detection, "^NETWORK"), "CRITICAL",
    match(detection, "Chrysalis|Payload"), "CRITICAL",
    match(detection, "USOShared|libtcc|Staging"), "HIGH",
    1==1, "HIGH"
)
| eval threat="Lotus Blossom APT - Chrysalis Backdoor Campaign (Feb 2026)"
| eval response="IMMEDIATE: Isolate host. Memory capture. Full IR engagement."
| table _time, host, User, severity, detection, threat, EventCode, sourcetype, target_path, dest_ip_field, dest_domain, DestinationPort, hash_field
| sort - severity, _time
```

Search 39 — EXECUTIVE NETWORK IOC DASHBOARD: Campaign network activity overview

Summary statistics of all network IOC matches for SOC leadership reporting.

Severity: INFO | **MITRE:** Multiple | **Data Sources:** All indexed data

SPL — Search 39

```
index=*
    (
        "45.76.155.202" OR "45.77.31.210" OR "59.110.7.32" OR "61.4.102.97"
        OR "95.179.213.0" OR "124.222.137.114" OR "45.32.144.255" OR "160.250.93.48"
        OR "103.159.133.178"
        OR "skycloudcenter.com" OR "wiresguard.com" OR "cdncheck.it.com"
        OR "temp.sh" OR "self-dns.it.com" OR "safe-dns.it.com" OR "cloudtrafficservice.com"
    )
| eval ioc_value=case(
    match(_raw, "45\.76\.155\.202"), "45.76.155.202",
    match(_raw, "45\.77\.31\.210"), "45.77.31.210",
    match(_raw, "59\.110\.7\.32"), "59.110.7.32",
    match(_raw, "61\.4\.102\.97"), "61.4.102.97",
    match(_raw, "95\.179\.213\.0"), "95.179.213.0",
    match(_raw, "124\.222\.137\.114"), "124.222.137.114",
    match(_raw, "45\.32\.144\.255"), "45.32.144.255",
    match(_raw, "160\.250\.93\.48"), "160.250.93.48",
    match(_raw, "103\.159\.133\.178"), "103.159.133.178",
    match(_raw, "(?i)skycloudcenter\.com"), "skycloudcenter.com",
    match(_raw, "(?i)wiresguard\.com"), "wiresguard.com",
    match(_raw, "(?i)cdncheck\.it\.com"), "cdncheck.it.com",
    match(_raw, "(?i)temp\.sh"), "temp.sh",
    match(_raw, "(?i)self-dns\.it\.com"), "self-dns.it.com",
    match(_raw, "(?i)safe-dns\.it\.com"), "safe-dns.it.com",
    match(_raw, "(?i)cloudtrafficservice\.com"), "cloudtrafficservice.com",
    1==1, "Other"
)
| eval ioc_type=if(match(ioc_value, "^\d+\.\d+\.\d+\.\d+$"), "IP Address", "Domain")
| stats count as total_events, dc(host) as unique_sources, earliest(_time) as first_seen, latest(_time) as last_seen, values(host) as observing_hos
    BY ioc_value, ioc_type
| eval first_seen=strftime(first_seen, "%Y-%m-%d %H:%M:%S")
| eval last_seen=strftime(last_seen, "%Y-%m-%d %H:%M:%S")
| eval hosting=case(
    ioc_value IN ("45.76.155.202", "45.77.31.210", "95.179.213.0", "45.32.144.255"), "Vultr VPS",
    ioc_value=="59.110.7.32", "Alibaba Cloud (port 8880)",
    ioc_value=="124.222.137.114", "Tencent Cloud (port 9999)",
    ioc_value=="160.250.93.48", "Unknown hosting",
    ioc_value=="103.159.133.178", "Unknown hosting",
    ioc_value=="61.4.102.97", "Unknown hosting",
    match(ioc_value, "wiresguard"), "WireGuard typosquat domain",
    1==1, "C2 domain"
)
| sort - total_events
| addcoltotals labelfield=ioc_value label="=== TOTAL ===" total_events, unique_sources
```

INCIDENT RESPONSE QUICK REFERENCE

If any CRITICAL detection fires:

1. IMMEDIATELY isolate the affected host from the network
2. Capture a full memory dump before any remediation
3. Preserve all Sysmon, Windows Event, and network logs
4. Check for lateral movement indicators on adjacent hosts
5. Search for all 9 C2 IPs and 7 C2 domains across all log sources
6. Verify no scheduled tasks or services reference Lotusblossom paths
7. Check Notepad++ installations for GUP.exe tampering
8. Engage full Incident Response team per organizational playbook

Key File Paths to Investigate:

%AppData%\Roaming\Bluetooth\log.dll (Chrysalis Loader)
%AppData%\Roaming\Bluetooth\BluetoothService.exe
%AppData%\Roaming\ProShow\ProShow.exe
%AppData%\Roaming\ProShow\load (no extension)
%AppData%\Roaming\Adobe\Scripts\script.exe
%AppData%\Roaming\Adobe\Scripts\alien.ini (C2 config)
C:\Windows\USOShared*.c, *.dll, *.exe
Any path containing libtcc.dll