Detecting Russian Threats to Critical Energy Infrastructure - Truesec

By Hjalmar Desmond

Published: 2026-02-09 · Archived: 2026-04-05 18:06:17 UTC

On December 29, 2025, a threat actor conducted a destructive cyberattack on the Polish electrical grid. The attack consisted of at least three separate events, including an attack targeting grid connection points (GCP) in the electric grid, and an attack targeting a combined heat- and powerplant (CHP) used to produce thermal energy.

This attack represents an escalation of the threat to critical infrastructure in the Nordics and Truesec assesses that now that Russia has crossed the threshold of conducting destructive cyberattacks on critical infrastructure, it can happen again and also in the Nordics.

It is consequently critical to find and eradicate other potential intrusions by this threat actor in environments belonging to critical infrastructure in the Nordics and the rest of Europe.

This blog is a summary of what we know about this threat actor, the tools and tactics used by them, and detection rules we have developed to assist in threat hunting and evicting them. Our aim is to assist defenders who want to protect critical infrastructure from this threat.

We are deeply indebted to the Polish CERT for their excellent report on the cyberattack on December 29. Much of what we know comes from this report.

Who is the Threat Actor

The report released by the Polish CERT attributes the attack to the Russian threat actor known as “Ghost Blizzard” or “DragonFly”. This attack represents the first known destructive cyberattack attributed to this group, and the first destructive cyberattack on critical energy infrastructure in a NATO country by a Russian cyber warfare unit.

The threat actor DragonFly is a cyber espionage and cyber warfare unit that is assessed to be part of Russia’s security service FSB Center 16. This group has been active since at least 2010 and appears to focus a lot of their activity on critical infrastructure sectors, including the energy, nuclear, commercial facilities, water, aviation, and critical manufacturing sectors.

Up until now DragonFly has never conducted any known destructive cyberattacks, even if they have repeatedly attempted to gain access to such networks, prepositioned backdoors, and prepared for destructive attacks.

The reason for this behavior is likely tied to the nature of cyber warfare. To conduct a successful destructive, cyberattack on a scale to impact an adversary at a critical point, usually requires extensive preparation. Artefacts from the Russian cyberattack on Ukraine on the eve of the Russian invasion of Ukraine in February 2022, shows that the operation had been prepared for at least a year ahead of the attack.

Russia's other main cyber warfare unit, the GRU threat actor known as Seashell Blizzard or "Sandworm" has conducted repeated cyberattacks in Ukraine in support of the Russian military operations, including repeated destructive attacks on the Ukrainian energy grid. Truesec assesses that DragonFly has not participated in these attacks because their main mission has been to maintain the capacity to strike at critical infrastructure in NATO countries, should the need arise.

That DragonFly has now been activated and conducted a destructive cyberattack against Polish critical infrastructure represents, in our assessment, a considerable escalation by Russia. Something that may have not received the attention it deserved, considering the effect this attack could potentially have had. Polish authorities claim that in a worst-case scenario, up to 500 000 people could have been left without electricity and heating.

Detection and Threat Hunting

Based on intelligence provided by the Polish CERT, Truesec conducted further analysis of the wiper malware used in the attack, as well as the associated malicious activities observed during the intrusion. The goal of this work was to provide additional detection mechanisms and enable effective threat hunting activities related to this campaign.

Wipers

The DynoWiper variant deployed against HMI systems was implemented in a specific manner, leveraging the Mersenne Twister (MT19937) pseudorandom number generator to produce random data used for overwriting files.

A heuristic approach to identifying binaries with potential wiper-like behavior is to look for constants associated with the Mersenne Twister algorithm in combination with Windows API calls commonly used for file enumeration and deletion. While this method is not specific to wiper malware, it can help surface suspicious binaries that merit further investigation.

Decompile: MersenneTwisterSeed - (Source.exe)

```
37    local_18 = 0;
38    local_28[0] = local_28[0] & 0xffffff00;
39    uVar2 = FUN_00405d4b();
40    param_1[1] = uVar2;
41    iVar3 = 1;
42    puVar4 = param_1 + 2;
43    do {
44      uVar2 = (uVar2 >> 0x1e ^ uVar2) * 0x6c078965 + iVar3;
45      iVar3 = iVar3 + 1;
46      *puVar4 = uVar2;
47      puVar4 = puVar4 + 1;
48    } while (iVar3 < 0x270);
49    *param_1 = 0x270;
50    iVar3 = 0;
51    do {
52      if (*param_1 == 0x270) {
53        FUN_00403ba0(param_1);
54      }
55      else {
56        if (0x4df < *param_1) {
57          MT19937_Refresh_State(param_1);
58        }
59      }
```

```
void __fastcall MT_Twister_Upper(int param_1)

{
  uint *puVar1;
  uint uVar2;
  int iVar3;

  iVar3 = 0x270;
  puVar1 = param_1 + 8;
  do {
    uVar2 = (puVar1[-1] ^ *puVar1) & 0x7fffffff ^ puVar1[-1];
    puVar1[0x26f] = -((uVar2 & 1) != 0) & 0x9908b0df ^ puVar1[0x18c] ^ uVar2 >> 1;
    iVar3 = iVar3 + -1;
    puVar1 = puVar1 + 1;
  } while (iVar3 != 0);
  return;
}
```

It should be noted that this approach may produce false positives, such as antivirus software or legitimate disk utility tools. Therefore, any binaries that match these heuristics should be analyzed in context to understand why such functionality exists on the system, especially if found on a system within an OT network.

```
import "pe"

rule possible_wiper_using_mersenne

{

    meta:

        description = "Windows PE < 500 KB containing MT19937 constants and wiper-like imports"
```

```
        date = "2026-02-02"
        author = "Nicklas Keijser"
        hash1 = "60c70cdcb1e998bffed2e6e7298e1ab6bb3d90df04e437486c04e77c411cae4b"
        hash2 = "835b0d87ed2d49899ab6f9479cddb8b4e03f5aeb2365c50a51f9088dcede68d5"
        hash3 = "65099f306d27c8bcdd7ba3062c012d2471812ec5e06678096394b238210f0f7c"
        hash4 = "d1389a1ff652f8ca5576f10e9fa2bf8e8398699ddfc87ddd3e26adb201242160"
    strings:
        $const = { 65 89 07 6C }
        $twist = { DF B0 08 99 }
        $mask7f = { FF FF FF 7F }
    condition:
        pe.is_pe and
        pe.imports( "kernel32.dll" , "GetLogicalDrives" ) and
        pe.imports( "kernel32.dll" , "FindFirstFileW" ) and
        pe.imports( "kernel32.dll" , "DeleteFileW" ) and
        pe.imports( "kernel32.dll" , "FindNextFileW" ) and
        pe.imports( "kernel32.dll" , "SetFileAttributesW" ) and
        filesize < 500KB and
        ($const and $twist and $mask7f) and
        (
          pe.number_of_signatures == 0 or
          (
            pe.number_of_signatures > 0 and
            not for any i in (0 .. pe.number_of_signatures - 1) :
              (
                pe.signatures [ i ] .issuer matches /Microsoft/i or
```

```
            pe.signatures [ i ] .subject matches /Microsoft/i
        )
    )
  )
}
```

The RTU Wiper

The destructive activities observed on RTU devices were performed by overwriting the firmware with a deliberately corrupted ELF binary. The file's entry point consists entirely of 0xFF bytes, rendering the firmware nonfunctional and effectively disabling the device.

This behavior can be detected by identifying ELF binaries where the entry point contains only 0xFF bytes. While this detection method is relatively simple and could be bypassed by a more sophisticated implementation, a match on this condition strongly indicates malicious intent, as such content would not be expected in a legitimate firmware image.

```
import "elf"

rule ELF_entrypoint_at_least_64_FF {

  meta:

    description = "ELF file with just FF at entry point"

    date = "2026-02-02"

    author = "Nicklas Keijser"

  condition:

    uint32(0) == 0x464c457f and

    for all i in (0..63) :

      (uint8(elf.entry_point + i) == 0xFF)

}
```

This rule just looks for FF at the entry point of the ELF file; this is rather trivial to bypass but if there is a match on this behavior, it is certain that the intention is malicious.

Threat Hunting

Prior to deploying the wiper malware, the threat actor performed multiple staging and preparation activities, primarily involving the movement and copying of files over SMB. These actions can be identified through structured threat hunting queries and, when detected, should be analysed in context to determine their purpose and origin.

Enable Automatic Administrative Shares

The attacker enabled automatic administrative shares to facilitate file movement and remote access.

Command:

“powershell.exe New-ItemProperty -Path ‘HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters’ -Name ‘AutoShareWks’ -Value 1 -PropertyType DWord -Force”

“powershell.exe New-ItemProperty -Path ‘HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters’ -Name ‘AutoShareServer’ -Value 1 -PropertyType DWord -Force”

Threat Hunting Query (User Input):

```
 DeviceProcessEvents
| where FileName in~ ("powershell.exe","pwsh.exe")
| where ProcessCommandLine has_all (@"LanmanServer\Parameters","New-ItemProperty") and ProcessCommandLine has_ar
```

Threat Hunting Query (System Output):

```
 DeviceRegistryEvents
| where ActionType == "RegistryValueSet"
| where RegistryKey endswith @"\LanmanServer\Parameters"
| where RegistryValueName has_any ("AutoShareWks","AutoShareServer")
| where RegistryValueData == 1
```

Restart of the SMB Service:

To activate the changes made to the LanmanServer configuration, the SMB service was restarted.

Command:

```
"powershell.exe Get-Service LanmanServer|Restart-Service -Verbose -Force"
```

Threat Hunting Query:

```
 DeviceProcessEvents
| where FileName in~ ("Powershell.exe","pwsh.exe")
| where ProcessCommandLine has_all ("LanManServer","Restart-Service")
```

Allow Inbound SMB Traffic on the Firewall

The attacker added a firewall rule allowing inbound TCP traffic on port 445 (SMB), using a misleading rule name to reduce suspicion.

Command:

“powershell.exe New-NetFirewallRule -Name ‘Microsoft Update’ -DisplayName ‘Microsoft Update’ -Protocol TCP -LocalPort 445 -Action Allow”

Threat Hunting Query:

```
 DeviceProcessEvents
| join kind=leftouter (DeviceInfo| where DeviceType == "Server" | where OSPlatform contains "Windows"| distinct
| where FileName in~ ("powershell.exe","pwsh.exe")
| where ProcessCommandLine has_all ("New-NetFirewallRule","Microsoft Update","-LocalPort 445","-Action Allow")
| where OSPlatform contains "windows"
```

Exfiltration Data via HTTP

Data exfiltration was performed using PowerShell’s **Invoke-RestMethod** cmdlet to upload files to a remote endpoint using HTTP POST requests.

Command:

Invoke-RestMethod -Uri -Method Post -InFile

Threat Hunting Query:

```
 DeviceNetworkEvents
| where InitiatingProcessCommandLine has_all ("Invoke-RestMethod","Post","InFile")
```

Conclusion and Recommendations

The detection mechanisms presented in this report are primarily behavior-based, focusing on identifying suspicious patterns and activities rather than relying on static indicators alone. This approach is well-suited for detecting destructive malware and preparatory activities where tooling, payloads, or infrastructure may change across intrusions.

It is important to emphasize that any alert or hit generated by these YARA rules or threat hunting queries must be investigated in context. While the behaviors described are highly indicative of malicious intent when observed together or in sensitive environments, certain legitimate tools, such as antivirus software, backup solutions, or disk utilities, may exhibit overlapping characteristics. Understanding the role of the affected system, the originating process, and the surrounding activity is therefore essential.

Based on validation and testing performed by Truesec, the expected rate of false positives for these detections is very low, estimated at less than 1 in 100,000 executions. This makes the provided rules and queries well-suited for proactive threat hunting, particularly in OT and critical infrastructure environments, where such behaviors are uncommon and should be treated with heightened scrutiny.

Recommendations

Truesec recommends the following actions:

- Deploy the provided YARA rules and hunting queries in monitoring and threat hunting workflows, with prioritization on OT-adjacent systems, HMI platforms, and infrastructure servers
- Investigate all hits thoroughly, correlating results with system role, change history, user activity, and other telemetry before drawing conclusions
- Baseline normal behavior for PowerShell, SMB, and firmware update activities within the environment to further reduce the risk of misinterpretation
- Treat detections related to wiper-like behavior as high severity, as these activities are rarely benign and often indicate imminent or ongoing destructive actions
- Continuously refine and adapt detections as new intelligence becomes available, particularly in relation to evolving wiper techniques and pre-positioning activities
- By applying these detections in combination with contextual analysis and operational awareness, organizations can significantly improve their ability to identify and disrupt destructive attacks at an early stage.

If you have any further questions regarding detection, threat hunting queries or threat to critical infrastructure please contact Truesec for further discussions.

Annex: TTPs Used by Threat Actor

Below is an amalgamation of what we know of TTP used by the “DragonFly” threat actor. A main source of information is again the report by the Polish CERT, supported by other sources.

RECONNAISSENCE		
Gather Victim Org Information: Business	T1591.002	Collected open source information to identify relationships between

Relationships		organizations for targeting purposes.
Active Scanning: Vulnerability Scanning	T1595.002	Scanned targeted systems for vulnerable Citrix and Microsoft Exchange services.
Phishing for Information: Spearphishing Attachment	T1598.002	Used spearphishing with Microsoft Office attachments to enable harvesting of user credentials.
Phishing for Information: Spearphishing Link	T1598.003	Used spearphishing with PDF attachments containing malicious links that redirected to credential harvesting websites.
RESOURCE DEVELOPMENT		
Acquire Infrastructure: Domains	T1583.001	Registered domains for targeting intended victims.
Acquire Infrastructure: Virtual Private Server	T1583.003	Acquired VPS infrastructure for use in malicious campaigns.
Compromise Infrastructure: Server	T1584.004	Compromised legitimate websites to host C2 and malware modules.
Obtain Capabilities: Tool	T1588.002	Obtained and used tools such as Mimikatz, CrackMapExec, and PsExec.
Stage Capabilities: Drive-by Target	T1608.004	Compromised websites to redirect traffic and to host exploit kits.
INITIAL ACCESS		
Valid Accounts: Local Accounts	T1078.003	Login to a Fortinet device within a manufacturing sector enterprise
External Remote Services	T1133	Use of Fortinet edge devices and Outlook Web Access (OWA) to gain infrastructure access
Drive-by Compromise	T1189	Compromised targets via strategic web compromise (SWC) utilizing a custom exploit kit.
Exploit Public-Facing Application	T1190	Conducted SQL injection attacks, exploited vulnerabilities CVE-2019-19781 and CVE-2020-0688 for Citrix

		and MS Exchange, and CVE-2018-13379 for Fortinet VPNs.
Supply Chain Compromise: Compromise Software Supply Chain	T1195.002	Ghost Blizzard has placed trojanized installers for control system software on legitimate vendor app stores.
Phishing: Spearphishing Attachment	T1566.001	Ghost Blizzard has sent emails with malicious attachments to gain initial access.
EXECUTION		
Exploitation for Client Execution	T1203	Exploited CVE-2011-0611 in Adobe Flash Player to gain execution on a targeted system.
Scheduled Task/Job: Scheduled Task	T1053.005	Distribution of the wiper within the domain using a Scheduled Task
PERSISTENCE		
Scheduled Task/Job	T1053	Creation of scripts on FortiGate devices for administrator credential theft and configuration modification
Valid Accounts: Local Accounts	T1078.003	Use of local FortiGate VPN accounts to connect to compromised entities
Account Manipulation: Additional Local or Domain Groups	T1098.007	Added newly created accounts to the administrators group to maintain elevated access.
External Remote Services	T1133	Use of FortiGate VPN to connect to compromised entities
Create Account: Local Account	T1136.001	Created accounts on victims, including administrator accounts, some of which appeared to be tailored to each individual staging target.
Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder	T1547.001	Added the registry value ntdll to the Registry Run key to establish persistence.

PRIVILEGE ESCALATION		
Valid Accounts: Accounts	T1078.003	Use of an account with administrative privileges on the edge device
Access Token Manipulation	T1134	Credential theft from the LSASS Service Privilege escalation via a Process Token
DEFENSE EVASION		
Masquerading: Masquerade Account Name	T1036.010	Created accounts disguised as legitimate backup and service accounts as well as an email administration account.
Indicator Removal: Clear Windows Event Logs	T1070.001	Cleared Windows event logs and other logs produced by tools they used, including system, security, terminal services, remote services, and audit logs. The actors also deleted specific Registry keys.[15]
Indicator Removal: File Deletion	T1070.004	Deleted many of its files used during operations as part of cleanup, including removing applications and deleting screenshots.
Modify Registry	T1112	Modified the Registry to perform multiple techniques through the use of Reg.
Template Injection	T1221	Injected SMB URLs into malicious Word spearphishing attachments to initiate Forced Authentication.
File and Directory Permissions Modification	T1222	Modification of file permissions by the wiper
Domain or Tenant Policy Modification: Group Policy Modification	T1484.001	Distribution of the wiper within the domain via modification of the “Default Domain Policy” GPO

Server Software Component: Web Shell	T1505.003	Use commonly created Web shells on victims' publicly accessible email and web servers, which they used to maintain access to a victim network and download additional malicious files. [15]
Impair Defenses: Disable or Modify System Firewall	T1562.004	Disabled host-based firewalls. The group has also globally opened port 3389.
Impair Defenses: Disable or Modify Network Device Firewall	T1562.013	Modification of FortiGate device configuration
Hide Artifacts: Hidden Users	T1564.002	Modified the Registry to hide created user accounts.
CREDENTIAL ACCESS		
OS Credential Dumping: Security Account Manager	T1003.002	Dropped and executed SecretsDump to dump password hashes.
OS Credential Dumping: NTDS	T1003.003	Dropped and executed SecretsDump to dump password hashes. They also obtained ntds.dit from domain controllers.
OS Credential Dumping: LSA Secrets	T1003.004	Dropped and executed SecretsDump to dump password hashes.
Brute Force	T1110	Attempted to brute force credentials to gain access.
Password Cracking	T1110.002	Dropped and executed tools used for password cracking, including Hydra and CrackMapExec.
Forced Authentication	T1187	Gathered hashed user credentials over SMB using spearphishing attachments with external resource links and by modifying .LNK file icon resources to collect credentials from virtualized systems.[15][7]
Steal or Forge Kerberos Tickets	T1558	Creation of the Diamond Ticket
DISCOVERY		

System Network Configuration Discovery	T1016	Used batch scripts to enumerate network information, including information about trusts, zones, and the domain. Retrieval of the routing table and ARP cache System Network
Remote System Discovery	T1018	Enumeration of systems available on the network
System Owner/User Discovery	T1033	Used the command query user on victim hosts.
Network Service Discovery	T1046	Enumeration of services available on the network
Connections Discovery	T1049	Enumeration of network connections
Process Discovery	T1057	Enumeration of processes running on the system
File and Directory Discovery	T1083	Used a batch script to gather folder and file names from victim hosts.
Account Discovery: Domain Account	T1087.002	Used batch scripts to enumerate users on a victim domain controller.
Network Share Discovery	T1135	Identified and browsed file servers in the victim network, sometimes , viewing files pertaining to ICS or Supervisory Control and Data Acquisition (SCADA) systems.[15]
Local Storage Discovery	T1680	Creation by the wiper of a list of disks visible to the system
COLLECTION		
Data from Local System	T1005	Collected data from local victim systems.
Query Registry	T1012	Queried the Registry to identify victim information.
Data Staged: Local Data Staging	T1074.001	Created a directory named “out” in the user’s %AppData% folder and copied files to it.

Screen Capture	T1113	Performed screen captures of victims, including by using a tool, scr.exe (which matched the hash of ScreenUtil).
Email Collection: Remote Email Collection	T1114.002	Accessed email accounts using Outlook Web Access.
Archive Collected Data	T1560	Compressed data into .zip files prior to exfiltration.
Data from Configuration Repository: Network Device Configuration Dump	T1602.002	Dumping firewall device configuration
COMMAND AND CONTROL		
Application Layer Protocol: File Transfer Protocols	T1071.002	Used SMB for C2.
Proxy	T1090	Use of reverse SOCKS Proxy and the Tor Network
Ingress Tool Transfer	T1105	Downloading tools from Dropbox
Remote Access Tools: Remote Desktop Software	T1219.002	Use of RDP to connect to devices in the internal network
Hide Infrastructure	T1665	Use of compromised infrastructure for communication
EXECUTION		
Command and Scripting Interpreter	T1059	Used the command line for execution.
PowerShell	T1059.001	Used PowerShell scripts for execution.
Windows Command Shell	T1059.003	Used various types of scripting to perform operations, including batch scripts.
Python	T1059.006	Used various types of scripting to perform operations, including Python scripts. The group was observed installing Python 2.7 on a victim.

Permission Groups Discovery: Domain Groups	T1069.002	Used batch scripts to enumerate administrators and users in the domain.
User Execution: Malicious File	T1204.002	Used various forms of spearphishing in attempts to get users to open malicious attachments.
System Services: Service Execution	T1569.002	Execution of commands using the PsExec tool
EXFILTRATION		
Exfiltration Over Web Service	T1567	Exfiltration of stolen data via HTTP o the attacker-controlled servers
Exfiltration Over Web Service: Exfiltration Over Webhook	T1567.004	Transmission of script execution results to a Slack channel
IMPACT		
Data Destruction	T1485	File corruption by the wiper
Inhibit System Recovery	T1490	Change of IP addressing on compromised devices
System Shutdown/Reboot	T1529	Device shutdown performed by the wiper

Sources

1. https://cert.pl/uploads/docs/CERT_Polska_Energy_Sector_Incident_Report_2025.pdf
2. https://www.sophos.com/en-us/research/resurgent-iron-liberty-targeting-energy-sector
3. https://docs.broadcom.com/doc/dragonfly_threat_against_western_energy_suppliers
4. https://vblocalhost.com/uploads/VB2021-Slowik.pdf
5. https://www.cisa.gov/news-events/cybersecurity-advisories/aa20-296a#revisions
6. https://www.cisa.gov/news-events/alerts/2018/03/15/russian-government-cyber-activity-targeting-energy-and-other-critical-infrastructure-sectors
7. https://docs.broadcom.com/doc/dragonfly_threat_against_western_energy_suppliers

---