Cold Reality: Impact of FrostyGoop Modbus ICS Malware Attacks on Connected OT Systems

Kyle O'Meara, Principal Adversary Hunter
Carolyn Ahlers, Principal Malware Analyst
Matt Pahl, Detection Technical Lead

AGENDA

1. BACKGROUND
2. FROSTYGOOP ICS MODBUS MALWARE
3. INVESTIGATING THE INCIDENT
4. THREAT LANDSCAPE RESEARCH
5. RECOMMENDATIONS
6. THREAT INTEL > DETECTIONS

FROSTYGOOP ICS MALWARE

9th
known ICS malware

Dragos discovered FrostyGoop binaries in April 2024.

1st
Modbus ICS malware that causes effects on ICS devices

FrostyGoop interacts directly with industrial control systems (ICS) using Modbus TCP over port 502.

ATTACK ON CONNECTED ENERGY SYSTEMS

What happened?

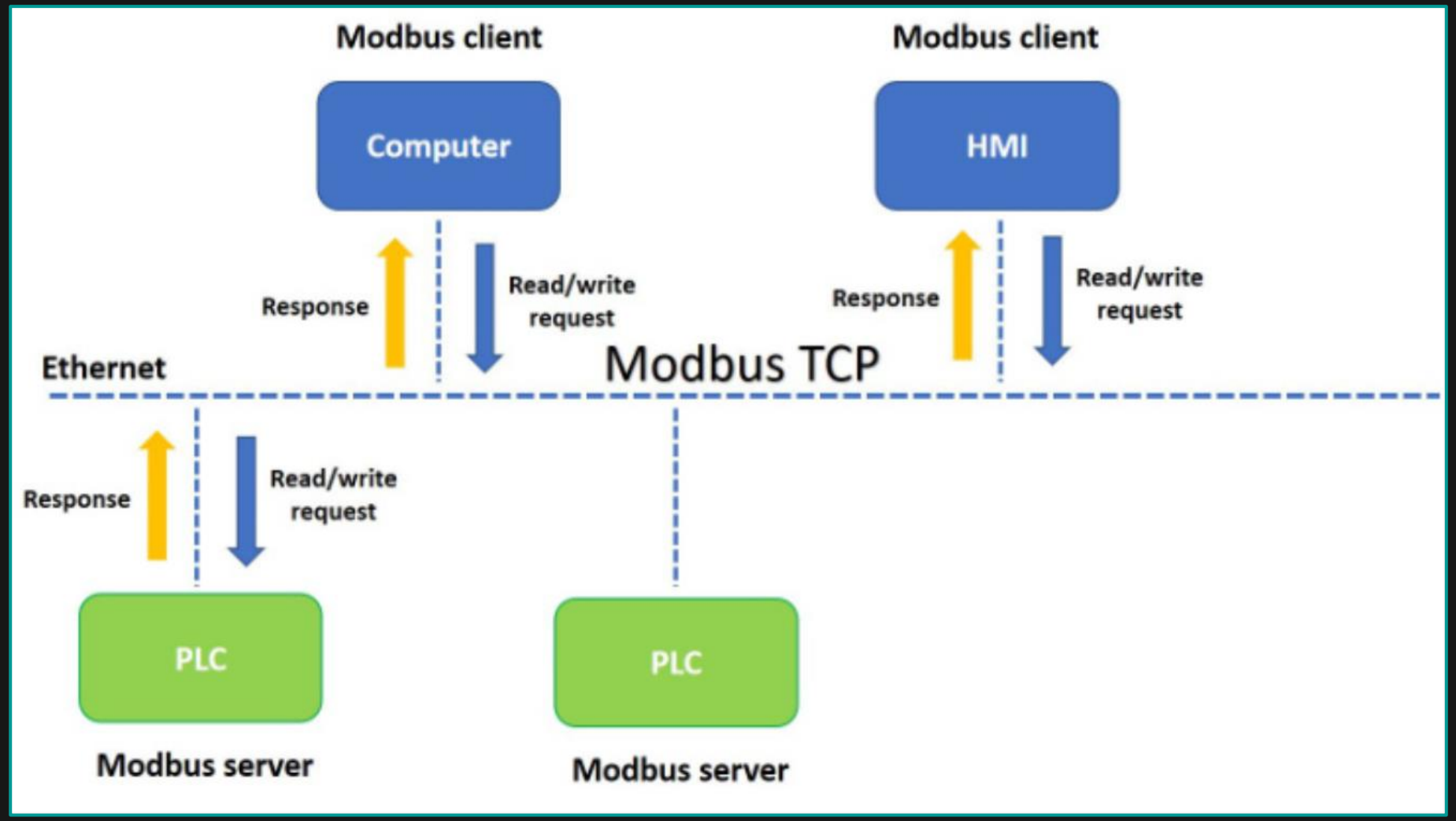
In January 2024, during sub-zero temperatures, a cyber attack disrupted the energy supply for central heating in more than 600 apartment buildings in Ukraine.

- The Cyber Security Situation Center (CSSC), part of the Security Service of Ukraine, shared details about this incident with Dragos.
- Dragos assessed that FrostyGoop was likely used to facilitate this attack.
- Dragos also assessed that before the attack, FrostyGoop was used to target controllers where TCP port 502 was internet-accessible.

FROSTYGOOP MALWARE CAPABILITIES

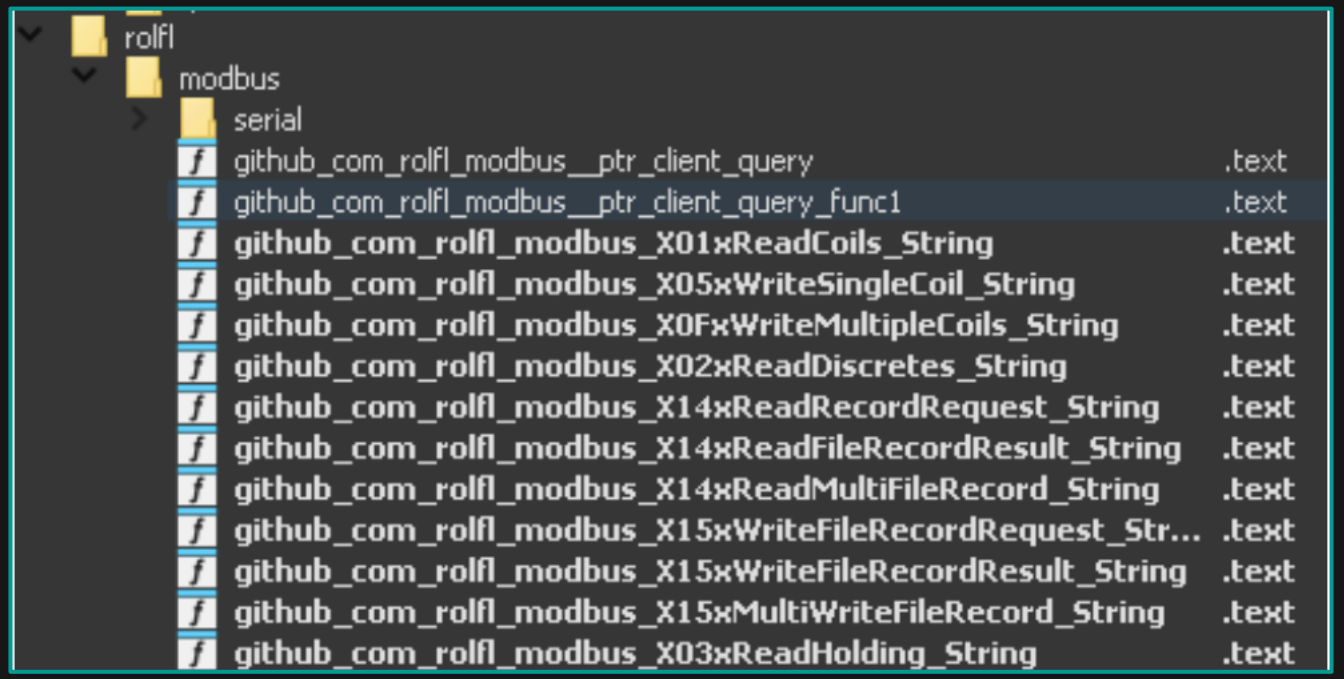
- Accept optional command line execution arguments
- Use a configuration file to specify target IP addresses and Modbus commands
- Communicate with ICS devices via Modbus TCP protocol
- Send Modbus commands to read or modify data on ICS devices
- Log output to console and/or JSON file

MODBUS TCP PROTOCOL



FrostyGoop

- Many binaries named modbus.exe
- Written in GoLang, use opensource libraries
  - github.com/rolfl/modbus/
  - github.com/goccy/go-json



FROSTYGOOP FUNCTIONALITY

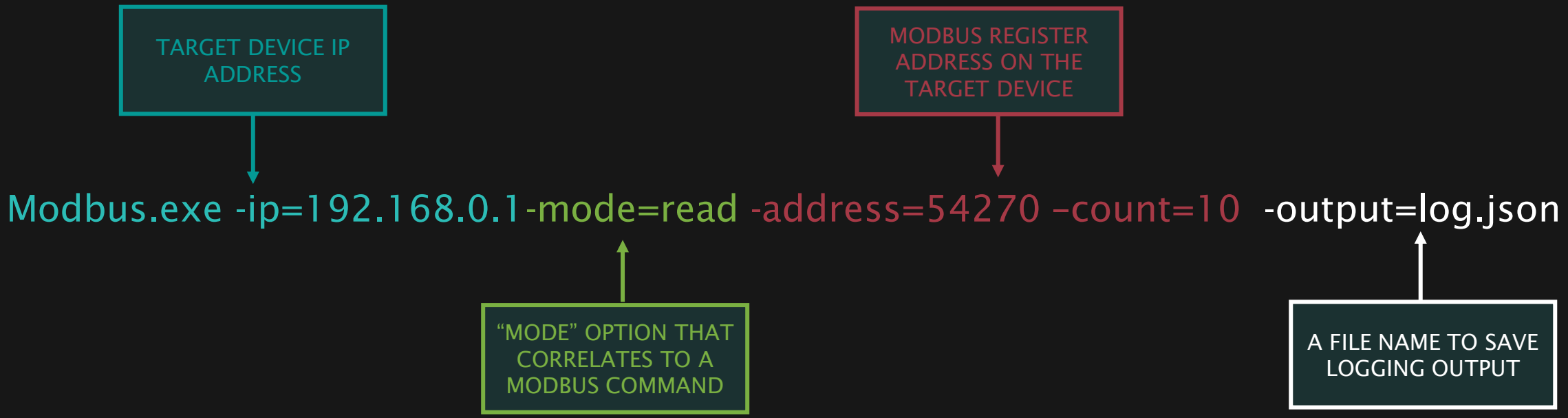
Information required to initiate a TCP connection and send Modbus commands to a victim ICS device can be specified in two ways:

Command Line Arguments

OR

JSON Configuration File

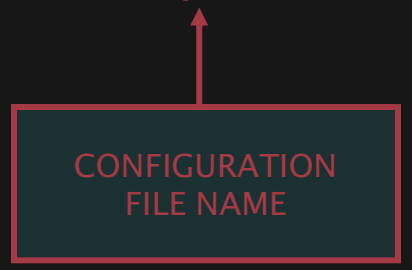
FROSTYGOOP COMMAND LINE ARGUMENTS



'read'	Command Code 3 'Read Holding Registers' - Read the value currently in a Modbus holding register (or contiguous block)
'write'	Command Code 6 'Write Single Holding Register' - Write a value to a holding register
'write-m'	Command Code 16 'Write Multiple Holding Registers' - Write a value to a block of contiguous registers

FROSTYGOOP JSON CONFIG

Modbus.exe -input-task=task.json




```
{
        "Iplist": ["192.168.0.1"
        ],
        "Tasks": [
                {
                        "Code": 3,
                        "Address": 54270,
                        "Count": 5,
                        "Value": 0
                },
                {
                        "Code": 3,
                        "Address": 54275,
                        "Count": 5,
                        "Value": 0
                },
                {
                        "Code": 6,
                        "Address": 54280,
                        "Count": 1,
                        "Value": 1
                },
                {
                        "Code": 16,
                        "Address": 54285,
                        "Count": 5,
                        "Value": 1
                }
        ]
}
```

Command Code 3 ‘Read Holding Registers’	Read the value currently in a Modbus holding register (or contiguous block)
Command Code 6 ‘Write Single Holding Register’	Write a value to a holding register
Command Code 16 ‘Write Multiple Holding Registers’	Write a value to a block of contiguous registers

FROSTYGOOP CONFIGURATION FILE

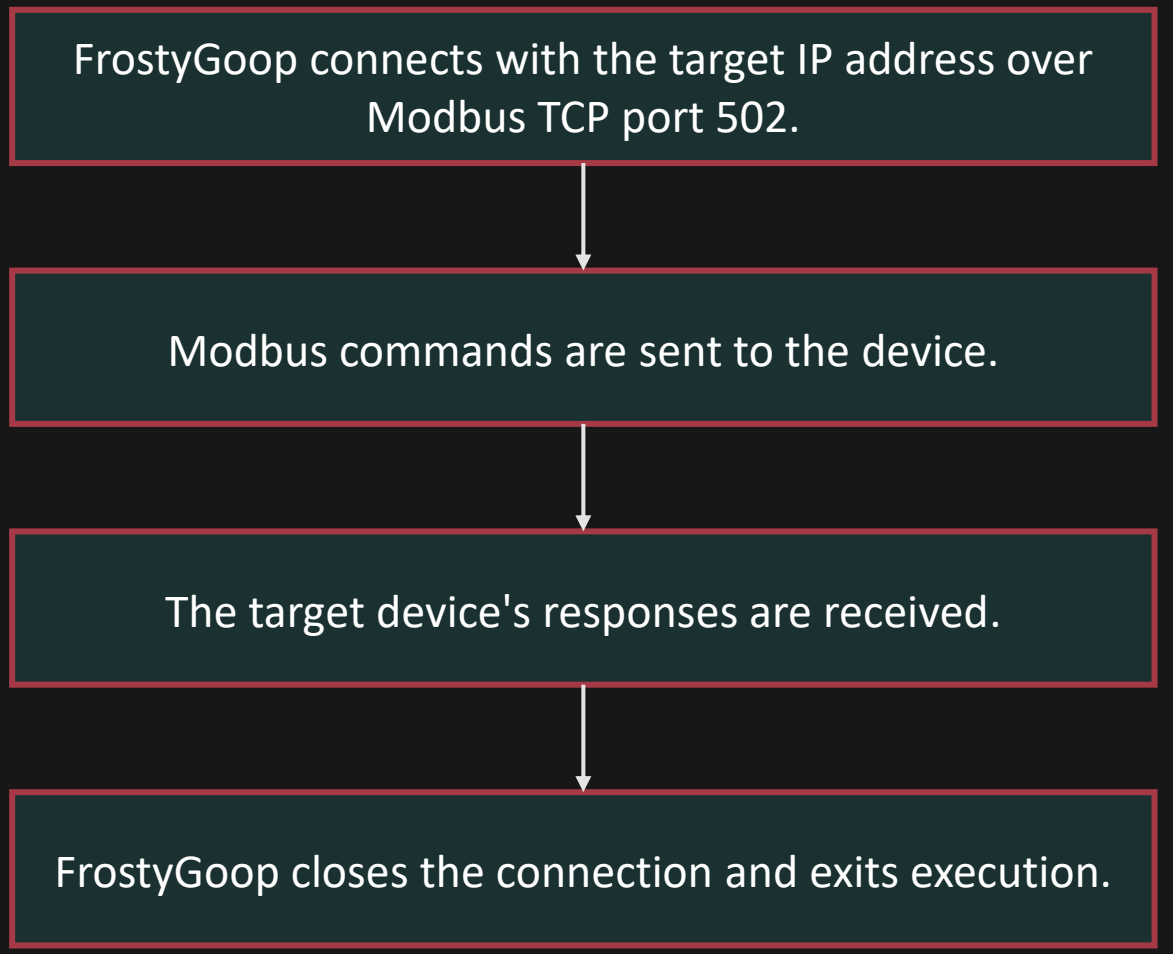
FrostyGoop accepts two different configuration files

Contains victim device information:

IP address, Modbus commands, Modbus register addresses

Specifies start time for Modbus TCP communications and delay time for executing Modbus commands.

MODBUS NETWORK TRAFFIC

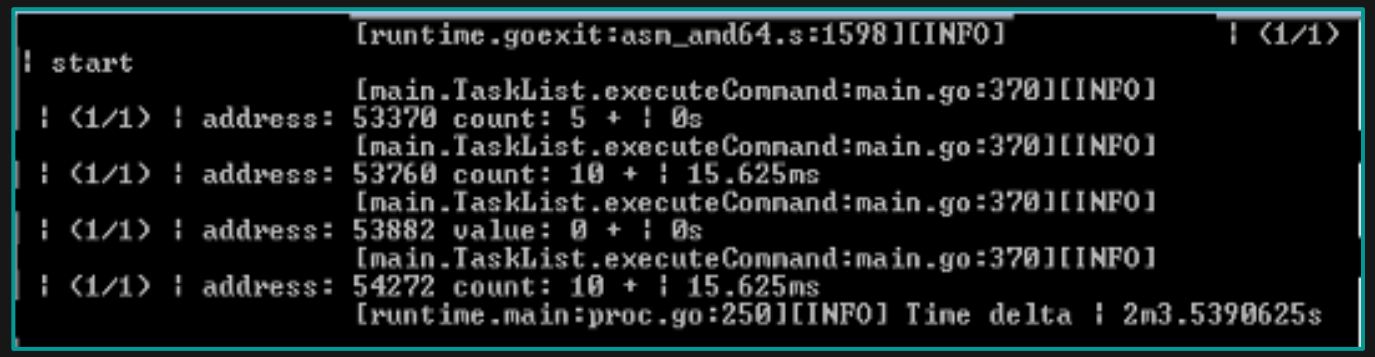


LOGGING CAPABILITIES

The FrostyGoop binaries log output from the Modbus TCP communications with the target IP address to a Windows console and a JSON file.

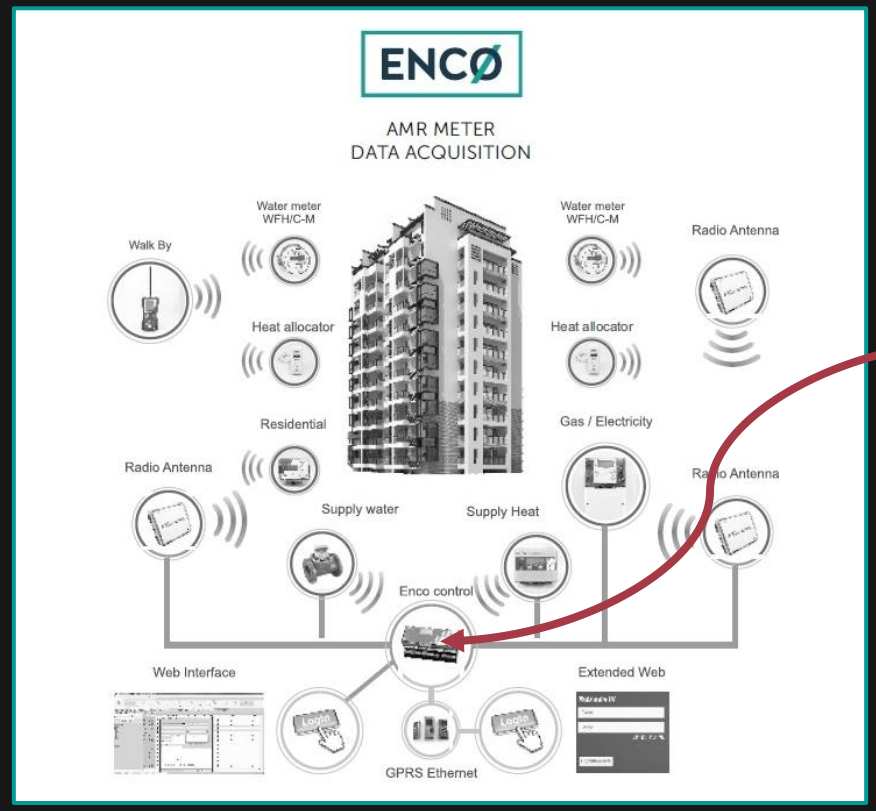
The FrostyGoop executable opens a Windows console upon execution. Below is an example of output logged to the console window during Modbus TCP communications with the target device.

If the argument for logging is specified when executing the binary, then the output is logged to a JSON file. Below is an example of the JSON log file.



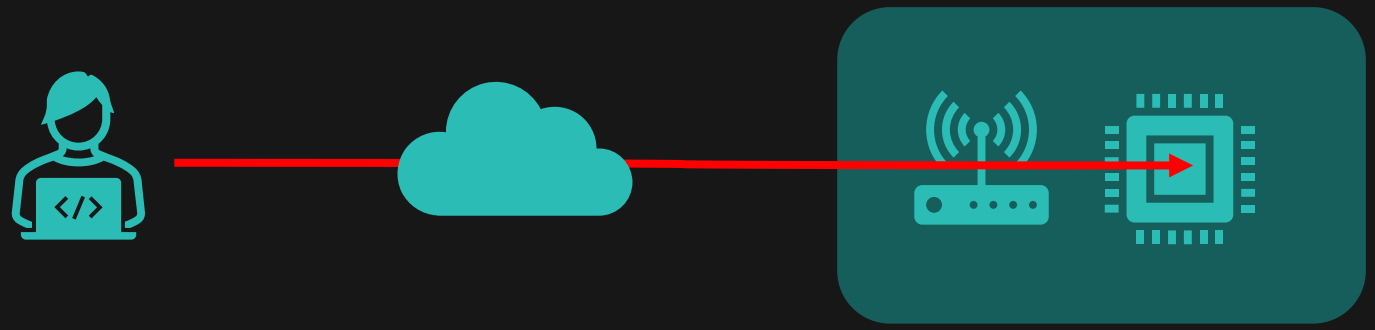
TASK_TEST.JSON

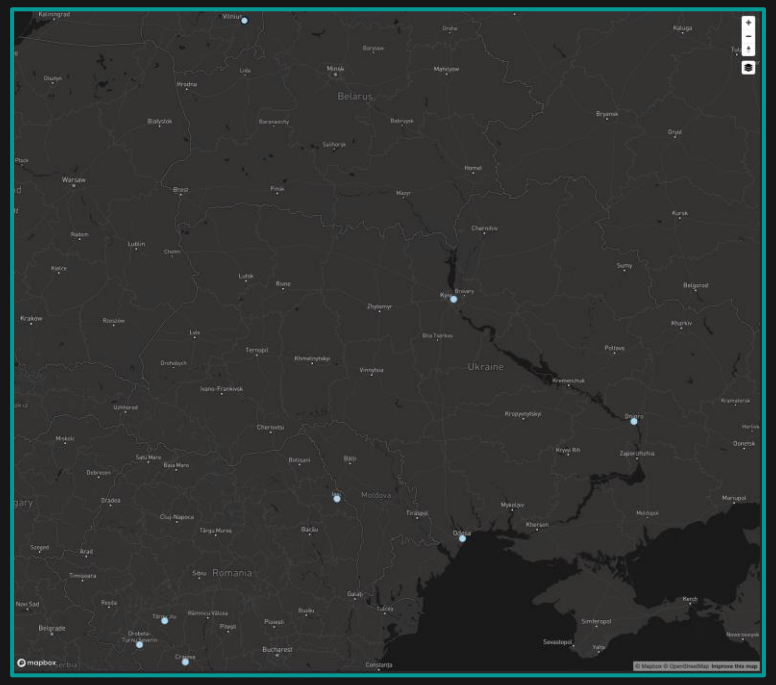
- Dragos discovered a sample of the configuration file named ‘task-test.json.’
- The IP address in the sample configuration file belongs to an ENCO control device.



- ENCO control devices are typically used “for process control in district heating, hot water, and ventilation systems” to monitor sensor parameters such as temperature, pressure, and insulation.

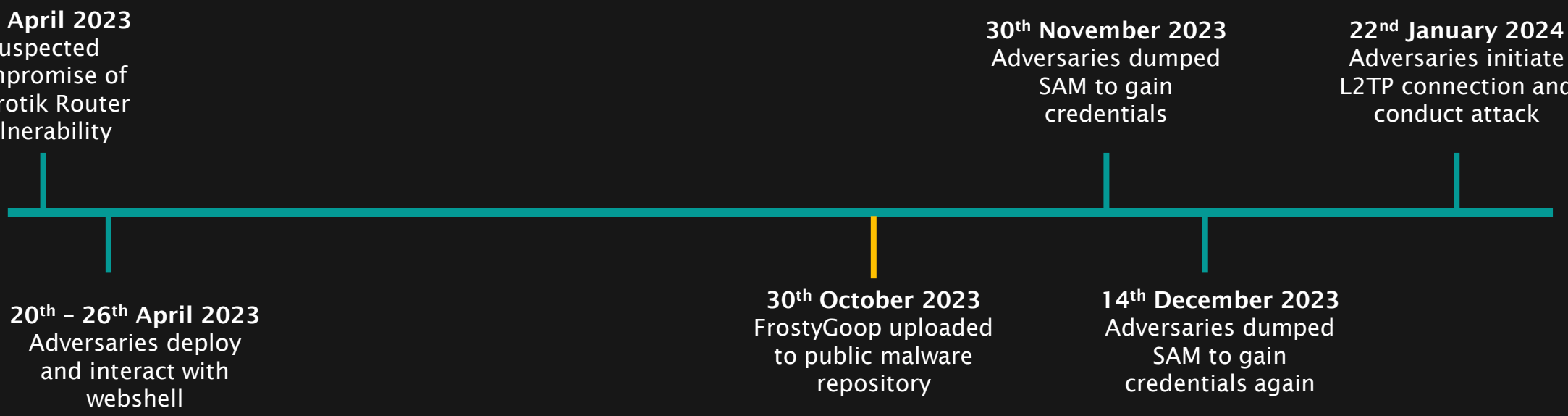
Theoretical Attack





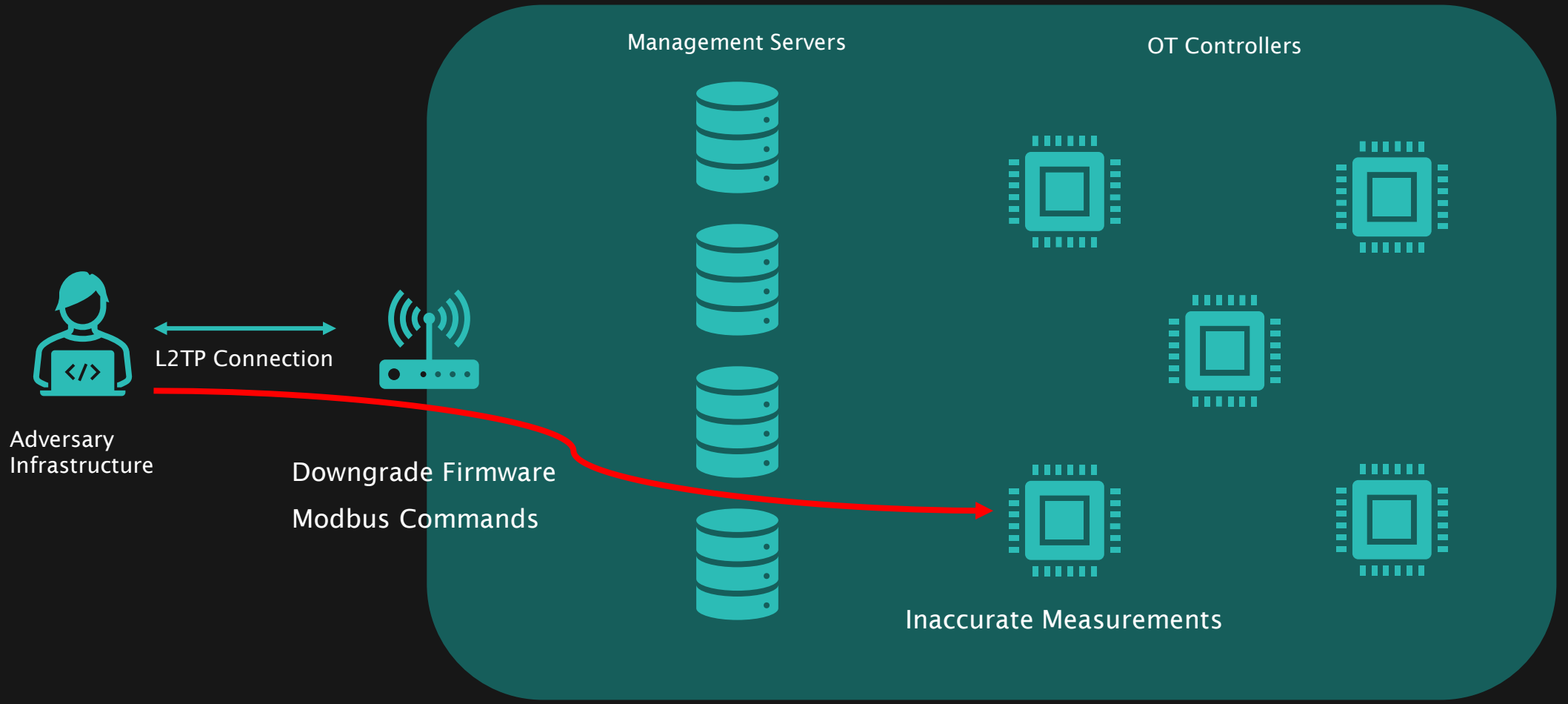

```
% nc -v 10.2.20.105 23
Connection to 10.2.20.105 port 23 [tcp/telnet] succeeded!
*=================================================================*
*             Enco control Telnet Server v1.00                    *
*=================================================================*

    Available Commands:
    ---------------------------
    ethr            - ethernet connection list
    gprs            - gprs connection list
    tcpconn         - tcp connections
    ipstat          - IP statistics
    icmpstat        - ICMP statistics
    tcpstat         - TCP statistics
    udpstat         - UDP statistics
    owire           - one wire temperature sensors list
    io              - inputs/outputs state
    outX=Y          - change output X=(0 or 1) state Y=(0 or 1)
    cport[=password,XXXX] - config port
    rst=password    - restart device
    ntp             - ntp correction
    help,?          - display this help
    exit,<Ctrl+C>   - disconnect
>
```

ATTACK TIMELINE



Details shared by The Cyber Security Situation Center (CSSC), a part of the Security Service of Ukraine (SBU)

The Attack Timeline

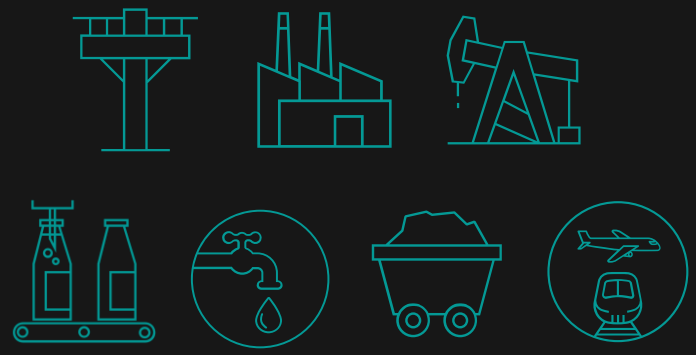


Details shared by The Cyber Security Situation Center (CSSC), a part of the Security Service of Ukraine (SBU)

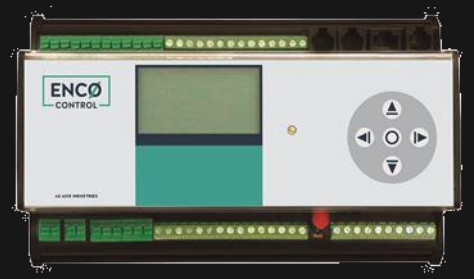
CONNECTED ICS DEVICE EXPOSURE

~46,000

Internet-exposed ICS devices communicating over Modbus



Modbus is used worldwide across industries.



~100

ENCO devices exposed

VULNERABLE PERIMETER DEVICES

- Exploiting vulnerable perimeter devices is not isolated to this event
- Threat groups and cybercriminals target edge devices for initial access

RECOMMENDATIONS

- Utilize network segmentation, least privileges, and user access control practices to prevent adversary lateral movement in network environments.
- Ensure effective visibility and monitoring of North-South traffic in ICS locations to identify assets, connection details, and compromise attempts and allow faster survey and response during a cyber incident.
- Implement access controls to critical ICS/OT systems and devices, including restricting Modbus TCP port 502 access.
- Monitor devices for new connections on port 502.
- Conduct network telemetry analysis for unusual interactions with devices over port 502.
- Immediately identify and ensure ICS/OT systems and devices are inaccessible from the public-facing Internet.
- Keep perimeter devices updated with the latest security patches.
- If devices need to be accessible, place them behind firewalls or in a DMZ.
- Implement multi-factor authentication for users, particularly privileged users, and for remote access.

POWER OF THE DRAGOS ECOSYSTEM

May 2024 – WorldView Subscribers Alerted of FrostyGoop binaries

June 2024 – WorldView Subscribers Alerted of Confirmed Attack on OT

June 2024 – Briefing for Dragos Customers

July 2024 – Public Announcement of FrostyGoop, 9th ICS Malware

OT CTI

PLATFORM & OT WATCH

NK

Dragos OT CTI Team

Dragos Platform

OT Watch Threat Hunting

Dragos Platform

Neighborhood Keeper

CTI Analysis & Investigation of FrostyGoop ICS Malware. Development of composite detections for Dragos Platform.

Modbus-Related Detections already in Platform would have detected FrostyGoop Behaviors. IOC's Deployed 1 Day after release via Knowledge Pack.

Dragos OT Watch Threat Hunters Launched a FrostyGoop Dashboard and Daily Hunts across the Customer Fleet.

Dragos Platform Continuously Monitoring for activity. Deployment of Composite Detections for FrostyGoop with Response Playbooks available.

Dragos Neighborhood Keeper Team Monitored Collective Defense data for related threats. None were found.

TRANSFORM THREAT INTEL INTO DETECTIONS

UNDERSTAND THE THREAT

BEHAVIOR, CAPABILITIES, INFRASTRUCTURE, INTENT

INITIAL ACCESS	EXECUTION	PERSISTENCE	PRIVILEGE ESCALATION	EVASION	DISCOVERY	LATERAL MOVEMENT	COLLECTION	COMMAND & CONTROL	INHIBIT RESPONSE FUNCTION	IMPAIR PROCESS CONTROL	IMPACT
Data Historian Compromise	Change Operating System	Modify Program	Exploitation for Privilege Escalation	Change Operating Mode	Network Connection Enumeration	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command Line Interface	Module Firmware	Hooking	Exploitation for Evasion	Network Sniffing	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Modify Parameter	Denial of Control
Engineering Workstation Compromise	Execution Through API	Project File Infection		Indicator Removal on Host	Remote System Discovery	Lateral Tool Transfer	Detect Operating System	Standard Application Layer Protocol	Block Command Message	Module Firmware	Denial of View
Exploit Public-Facing Application	Graphical User Interface	System Firmware		Masquerading	Remote System Information Discovery	Program Download	I/O Image		Block Reporting Message	Spoof Reporting Message	Loss of Availability
Exploitation of Remote Services	Hooking	Valid Accounts		Rootkit	Wireless Sniffing	Remote Services	Man in the Middle		Block Serial COM	Unauthorized Command Message	Loss of Control
Internet Accessible Device	Modify Controller Tasking			Spoof Reporting Message		Valid Accounts	Monitor Process State		Data Destruction		Loss of Productivity & Revenue
Remote Services	Native API						Point & Tag Identification		Denial of Service		Loss of Protection
Replication Through Removable Media	Scripting						Program Upload		Detect Restart/ Shutdown		Loss of Safety
Rogue Master	User Execution						Screen Capture		Manipulate I/O Image		Loss of View
Spearfishing Attachment							Wireless Sniffing		Modify Alarm Settings		Manipulation of Control
Supply Chain Compromise									Rootkit		Manipulation of View
Wireless Compromise									Service Stop		Theft of Operational System
									System Firmware		

DATA SOURCES: DRAGOS THREAT INTELLIGENCE, OSINT RESEARCH, THIRD-PARTY THREAT INTELLIGENCE

OPERATIONAL CONSTRAINTS

PIVOT AGAINST BEHAVIORS & OPERATE WITHIN PLATFORM CAPABILITIES

Type

Indicator
Configuration
Modeling
Threat Behavior

Complexity

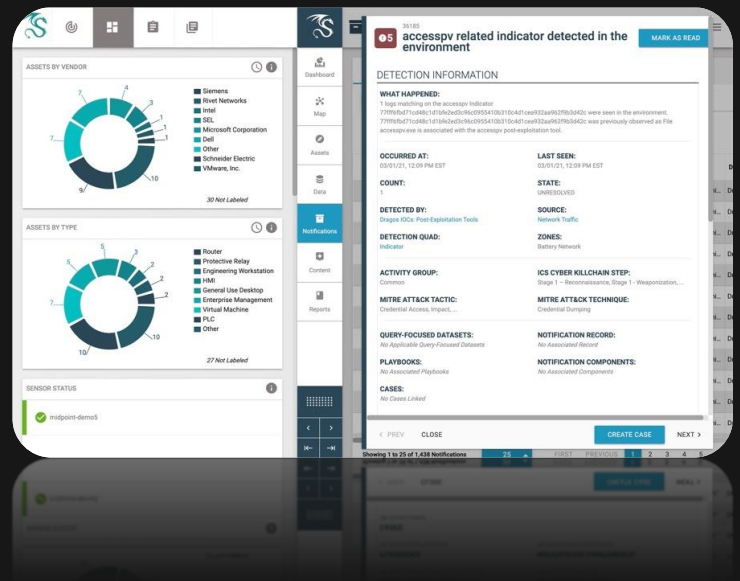
Atomic
Composite

Telemetry

Network Monitoring
Host Logs

DETECTIONS ARE CODIFIED IN THE DRAGOS PLATFORM

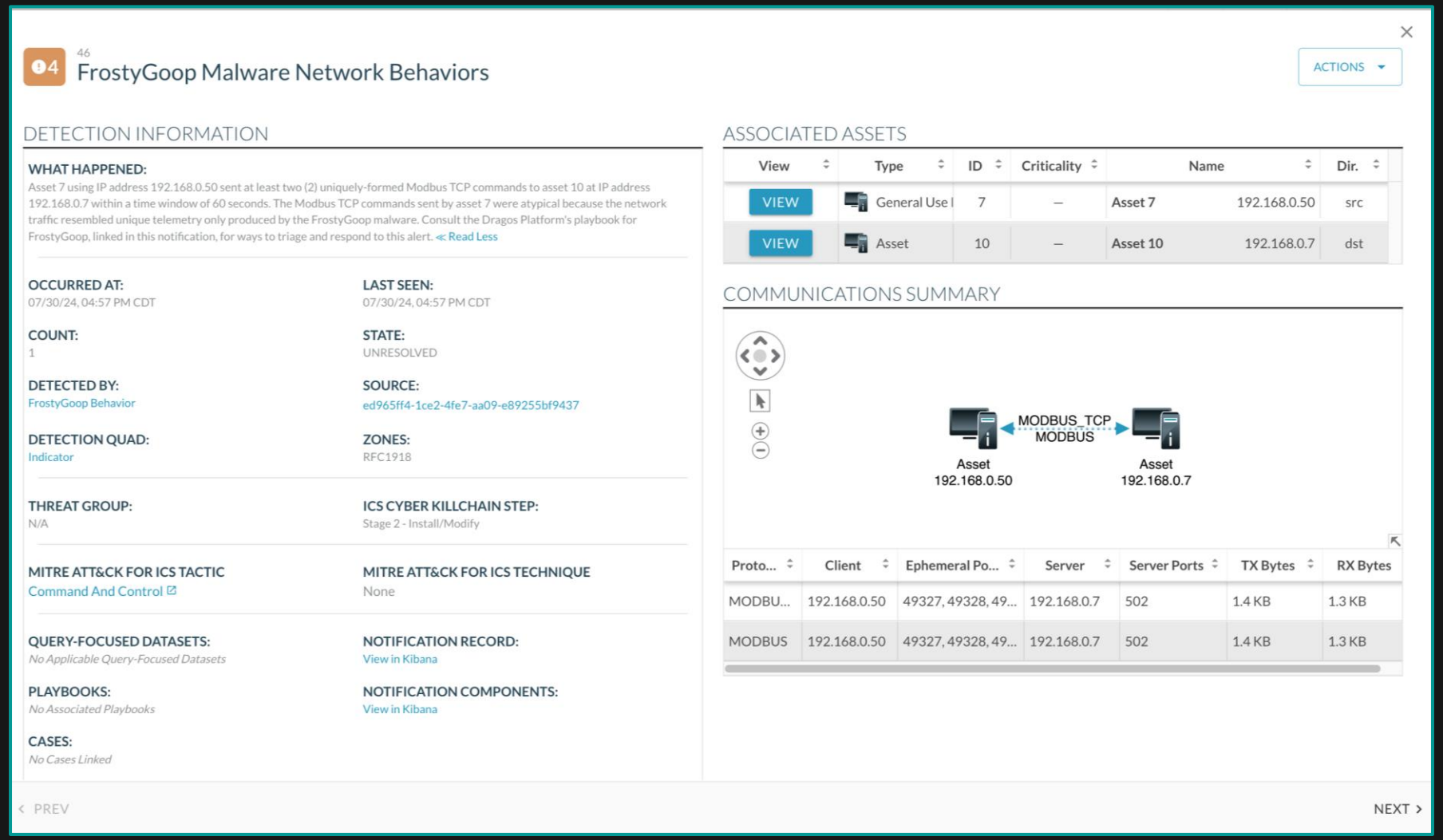
KNOWLEDGE PACKS RELEASED REGULARLY WITH NEW THREAT INTELLIGENCE-DRIVEN DETECTIONS



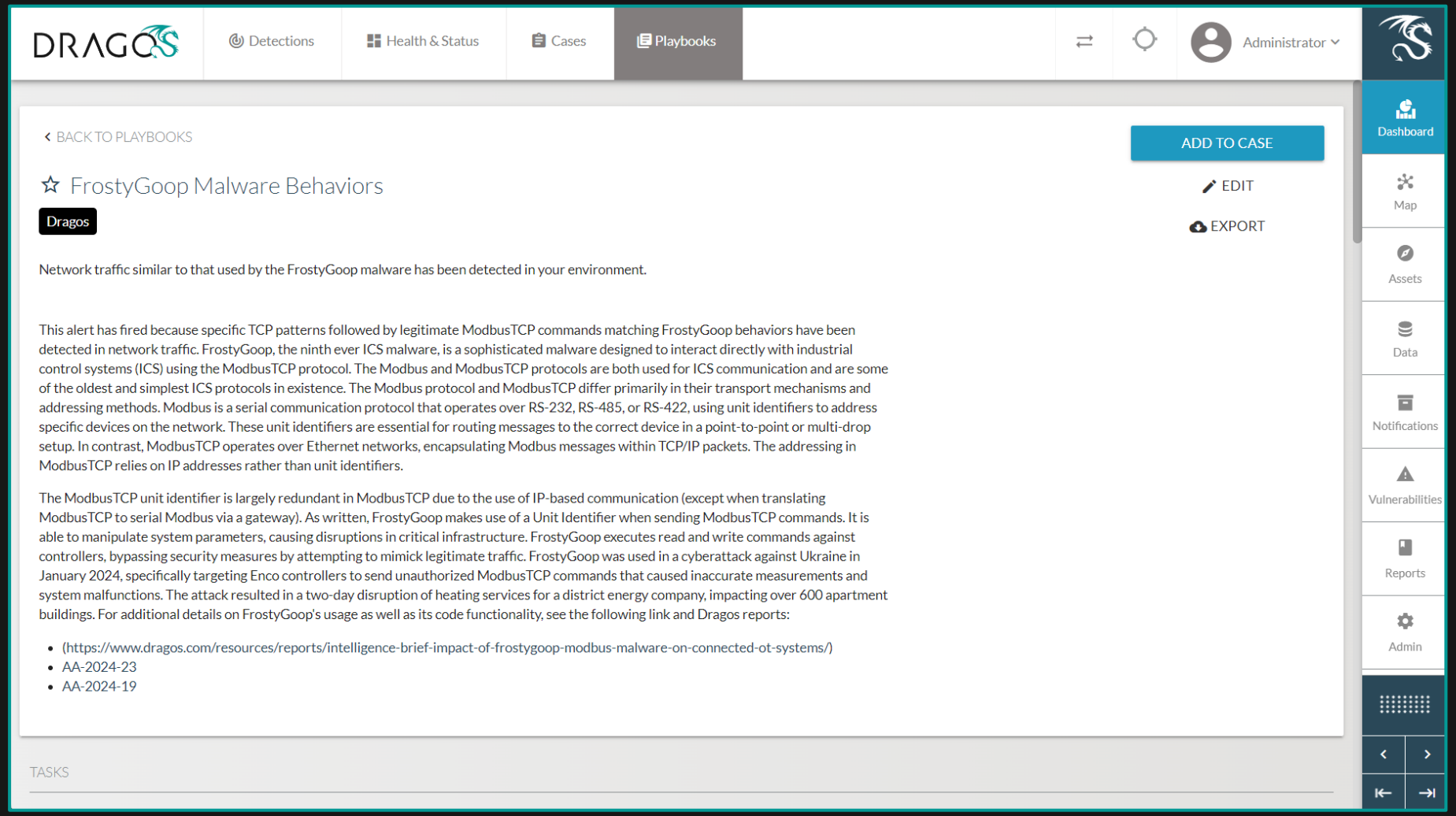
DRAGOS PLATFORM – EXISTING MODBUS DETECTIONS

SID/Rule	Analytic Name	Description	Knowledge Pack
a0ddb920-0adc-4d01-9b3d-21414ef28607	Modbus Command Force Listen Only Mode	Modbus command to put device into Force Listen Only Mode, making the device unresponsive to commands. It will only respond after power up. This can be used maliciously to effectively disable devices.	KP_Plus-7.0.X
f7a0af6b-fa88-4382-9232-f56525befcde	Modbus Command Restart Communications Option	Modbus command to force a device to restart, making it unresponsive until it reboots. There is some chance this could be used maliciously to disable devices.	KP_Plus-7.0.X
f41c99e6-cabf-46b7-9576-d2ac4676baa9	Modbus Exception	Modbus servers send exception codes to Modbus clients when a requested operation cannot be carried out. This characterization summarizes exception codes sent from a Modbus server.	KP_Plus-6.0.X
e8cbde89-aa3a-4093-8064-3a8ca08fbf4c	Modbus External Comms	External device communicating with an internal asset using the Modbus protocol. This is a major security concern, as ICS devices should not be controlled outside of the OT network.	KP-2020-11
15c07ad4-5d03-4c3b-8d2d-613d5ec45217	Modbus External Write	External device writing to an internal asset using the Modbus protocol. This is a major security concern, as ICS devices should not be controlled outside of the OT network.	KP-2020-11
3cc434cd-5086-454c-bbd4-6142b01a4623	Modbus Write Observed for First Time	Modbus traffic with a write function code seen for the first time to a specific host.	KP-2022-009
d323014b-abee-461b-a12f-641b8796070f	New ModbusTCP Detection	Monitors for new devices using the ModbusTCP protocol and generates events when activity is seen	KP-2020-11

FROSTYGOOP COMPOSITE DETECTION



FROSTYGOOP PLAYBOOK – DRAGOS PLATFORM



RECOMMENDATIONS

SANS

THE FIVE ICS CYBER SECURITY CRITICAL CONTROLS

01
ICS Incident Response Plan

02
Defensible Architecture

03
ICS Network Visibility & Monitoring

04
Secure Remote Access

05
Risk-based Vulnerability Management

QUESTIONS AND ANSWERS