

TECHNIQUE T875: CHANGE PROGRAM STATE – EXECUTION

CyOTE Use Case(s)	MITRE ATT&CK for ICS® Tactic
Alarm Logs, HMI, Remote Login	Execution
Data Sources	
Potential Data Sources	Data Historian points, Application Logs, OS Stack Logs, Packet Captures, Network Protocol Analysis
Historical Attacks	Triton Attack at Petro Rabigh ¹

TECHNIQUE DETECTION

The Change Program State technique² (Figure 1) may be detected when a device’s program state is changed without warning or reason.

To augment commercial sensor gaps, the CyOTE program has developed capabilities such as Proof of Concept tools³ and Recipes⁴ for asset owners and operators (AOO) to identify indicators of attack for techniques like Change Program State within their operational technology (OT) networks. Referencing CyOTE Case Studies⁵ of known attacks, AOOs in both small and large organizations can utilize CyOTE’s Use Case analyses to tie operational anomalies and observables to cyber-attack campaigns resulting in ever-decreasing impacts.

PERCEPTION: OBSERVABLES FROM HISTORICAL ATTACKS

The Change Program State technique was used in the Triton attack at Petro Rabigh in 2017.⁶ In this attack, the following observables were identified:

- Increased internet traffic
- Increased DMZ traffic between information technology (IT) and OT networks

¹ MITRE, *Software: Triton, TRISIS, HatMan*, <https://collaborate.mitre.org/attackics/index.php/Software/S0013>

² MITRE ATT&CK for ICS, T875: Change Program State, <https://collaborate.mitre.org/attackics/index.php/Technique/T0875>. Note that this technique has been deprecated, and its content merged into T858, Change Operating Mode: <https://collaborate.mitre.org/attackics/index.php/Technique/T0858>

³ A Proof of Concept tool is a representative implementation of a set of steps and methods for identifying techniques. A Proof of Concept tool is defined as a script(code) or using capabilities of existing tools (e.g., Splunk, Gravwell), to demonstrate the capability to identify adversarial activity for a selected technique. A Proof of Concept tool is not ready for implementation in an AOO’s environment as its major focus is to a specific instance (device, vendor, protocol, scenario) in order to prove a concept.

⁴ A Recipe is a set of steps and methods for identifying techniques. Recipes can be used to develop a Proof of Concept or operational tool in an AOO’s OT environment.

⁵ Visit <https://inl.gov/cyote/> for all CyOTE Case Studies.

⁶ <https://www.eenews.net/stories/1060123327>

Disclaimer: Past occurrences are not guaranteed to occur in future attacks.

COMPREHENSION

In the Triton attack at Petro Rabigh, the adversary first gained access through an engineering workstation to map the network; once they gained control of the workstation, they moved through the network and deployed the malware, changing program states and device logic to issue malicious command messages that shut down part of the plant.⁷ By understanding the nature and possible origins of this attack, as well as how the adversary used the Change Program State technique to execute the attack, an AOO can better comprehend how this technique is used with others and enhance their capabilities to detect attack campaigns using this technique and decrease an attack's impacts.

CURRENT CAPABILITY

The CyOTE Recipe describes how to develop a capability that will regularly pull or receive logs from the data historian, collection server, or security information and event management (SIEM) to parse, analyze, and compare changes. This capability monitors for basic operational state changes (e.g., PLC main program is started, stopped, or reset). A state change log entry will trigger an alert for further evaluation. A configuration file is used to define which logs to pull and identify log entries containing potential observables. The capability's output provides statistics about observables, number of times triggered, which PLC caused the trigger, and the log where the entries were found.

POTENTIAL ENHANCEMENTS

Additional research is needed to monitor and compare data historian, collection server, or SIEM logs to identify potential operational state changes and integrate a calendar for maintenance windows (emergency or planned). The calendar will be used to reduce triggerable events during maintenance windows. If an observable is found, the calendar will be checked; if the observable occurred outside of maintenance, it is flagged as unauthorized, and an alert is generated. To further reduce alert generation, the process environment will be baselined by the capability, looking for traffic norms, command usage frequency, and other baselines.

ASSET OWNER DEPLOYMENT GUIDANCE

Deploying this capability in a continuously running state will require asset owners to configure data historian, collection server, or SIEM to send logs to this capability which will alert when log entries match patterns in the configuration file for potential state changes. If a log does not match, an alert is received. Alerts can be customized for output to a JSON log or a STIX 2.1 format.

AOOs can refer to the CyOTE Technique Detection Capabilities report (visit <https://inl.gov/cyote/>) for more information on the background and approach of CyOTE's technique detection capabilities.

⁷ CyOTE Case Study: Triton in Petro Rabigh. <https://inl.gov/wp-content/uploads/2021/09/Triton-CyOTE-Case-Study.pdf>

AOOs can also refer to the [CyOTE methodology](#) for more information on CyOTE’s approach to identifying anomalies in an OT environment, which, when perceived, initiates investigation and analysis to comprehend the anomaly.

Click for More Information

[CyOTE Program](#) || [Fact Sheet](#) || CyOTE.Program@hq.doe.gov

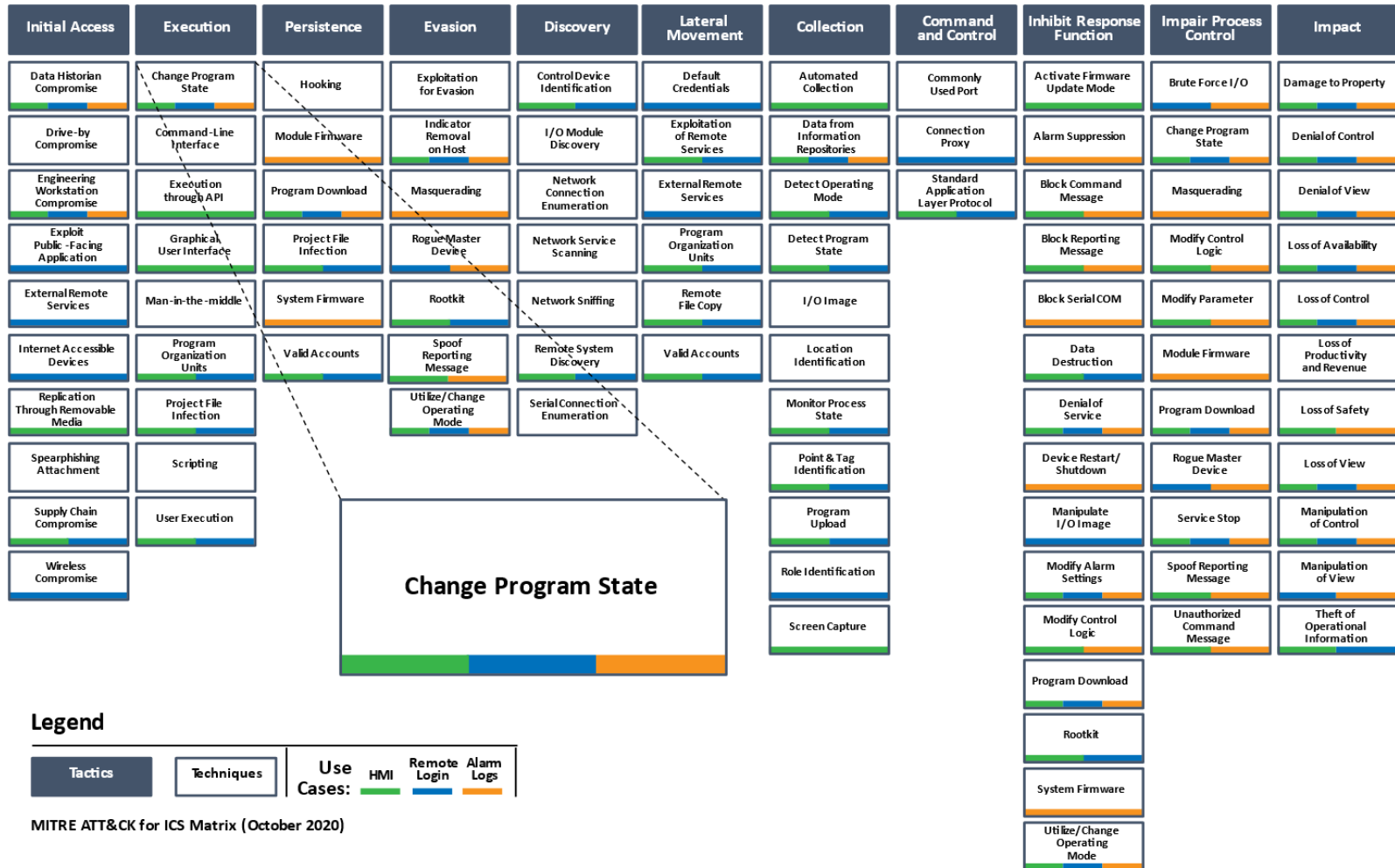


Figure 1: ICS ATT&CK Framework⁸ – Change Program State Technique – Execution

⁸ © 2020 The MITRE Corporation. This work is reproduced and distributed with the permission of The MITRE Corporation.