SCADA System Security: Accounting for Operator Error and Malicious Intent

Ryan Kilbride
Jeremy Straub
Eunjin Kim
SCADA System Security: Accounting for Operator Error and Malicious Intent
Ryan Kilbride, Jeremy Straub, Eunjin Kim
Department of Computer Science, University of North Dakota

Introduction
Supervisory control and data acquisition (SCADA) systems are becoming more and more commonplace in many industries today. Industries are making better use of software and large scale control systems to run efficiently, without the need for large amounts of oversight. Security is a particularly large issue with such systems, however. A human must still be involved to ensure smooth operation in the event of catastrophic system error, or unusual circumstances. Human involvement presents problems: operators could make mistakes, configure the system to operate sub-optimally or take malicious actions. This implementation of SCADA security aims to combat these problems.

Though there are many facets of SCADA security, including cryptography, network security, and adequate system administration, this implementation will be focused on software-based solutions. There are three areas of consideration: systems monitoring, intrusion/ anomaly detection and forensics, and counteraction of possible threats. Constraint satisfaction algorithms and use of an expert system are the main processes used.

System Monitoring
One of the weak areas of most process control systems is lack of monitoring information on the commands issued and the users making these commands. Separation of user accounts is seldom done and login attempts are often not recorded [3]. This security system, however, is highly dependent on real-time analysis of control processes. These processes begin with monitoring user logins, both over the internal SCADA network and external network. Data on a user’s incorrect logins, IP/device used to access the network, device history on the network, and user’s individual use of system commands will be assessed for the duration of a user session. A user’s trustworthiness can be inferred from this data and their deviation from past actions will help determine if problems may arise.

In addition to user monitoring, the system’s state will currently be assessed in case of failure to catch user-issued commands. This could be caused by a direct manipulation of devices on sight that bypass software systems entirely. Though the threat cannot be physically blocked, the system can counteract those actions if detected. These processes are shown below (Figure 1).

Command Validation and Forensics
The forensics unit is responsible for determining issues with user commands and device output. In the command validation, the subsystem will consult the knowledge base (KB) used by the expert system for determining outcomes of certain user processes. It may also be used to infer data about unknown system states. Figure 2 below shows the detailed user command validation process. The forensic unit also detects possible bypasses of the command network/ equipment failure. An overview is shown at right (Figure 3). Since these events are largely out of the system’s control, a human user will be notified.

Event Handling
Abnormal event handling is largely dependent on the particular SCADA system in use. Some systems may want to escalate certain situations to users with elevated permissions or require third-party validation. This system should be flexible enough to allow for these changes. In theory, this system should be compatible with a number of SCADA expert systems.

Future Work
The system is nearing its final stages of development, but there is still work to be done. The counter-action system for the particular implementation is midway through production. The system has yet to be assigned a working knowledge base and evaluated thoroughly. Future implementations of this system, for larger process control applications, may also be considered.

References

Background
Supervisory control and data acquisition (SCADA) systems are types of industrial control systems that monitor and issue commands over a networked interface [1]. These systems are often used in manufacturing, power generation, and many other industrial areas. SCADA systems allow a reduced number of on site employees and direct control to centralized locations. Users at these locations have control of many individual aspects of a system’s nature.

SCADA systems are commonly compromised of a local network system with multiple components. Servers and workstations, similar to a corporate IT network, are used by operators for system operation. These operators use Human-Machine Interfaces (HMIs) to control aspects of the system, as well as interpret system operations and real time variables [2].

---

**Figure 1. An overview of the System Monitor**
User input is duplicated and sent to both the process control system, where input will be processed into device commands, and the monitor, which will log relevant data on the commands. Similarly, the device monitor will monitor device output. Note the independence of the processed system commands and the data recorded.

**Figure 2. Forensics algorithm**
This figure shows a detailed look at the user input verification process. This process continues while a user is considered to be untrustworthy or until the user exits the system.

**Figure 3. Comparison of device output and user-issued commands**